



**Escola Universitària Politècnica
de Vilanova i la Geltrú**

UNIVERSITAT POLITÈCNICA DE CATALUNYA

MINIPROJECTE AUTOMATITZACIÓ INDUSTRIAL

TÍTOL: SISTEMES SCADA

AUTORS: Pedro Fernández Gómez

Nèstor Sorlí Martínez de Oraà

Jordi Ramon Arrufat Bordell

TITULACIÓ: Enginyeria Tècnica Industrial especialitat Electrònica Industrial

DIRECTOR: Pere Ponsa

DEPARTAMENT: ESAII

DATA: 11 de Desembre de 2003

TÍTOL: SISTEMES SCADA

COGNOMS: Sorlí Martínez de Oraà

NOM: Néstor

COGNOMS: Fernández Gómez

NOM: Pedro

COGNOMS: Arrufat Bordell

NOM: Jordi Ramon

TITULACIÓ: Enginyeria Tècnica Industrial

ESPECIALITAT: Electrònica Industrial

PLA: 95

DIRECTOR: Pere Ponsa

DEPARTAMENT: ESAII

QUALIFICACIÓ DEL MP

TRIBUNAL

PRESIDENT

Pere Ponsa

SECRETARI

Pere Ponsa

VOCAL

Pere Ponsa

DATA DE LECTURA: 15 de Desembre de 2003

Aquest Projecte té en compte aspectes mediambientals: ☐ Sí ☒ No

MINIPROJECTE AUTOMATITZACIÓ INDUSTRIAL

RESUM (màxim 50 línies)

La finalitat d'aquest miniprojecte és aconseguir fer un treball sobre sistemes SCADA, on poguem veure mecanismes per PC i mecanismes hardware per la realització d'aquest.

En primer lloc s'estudia que significa un sistema SCADA (Supervisory Control And Data Acquisition), quines prestacions, mòduls i necessitats bàsiques que ha de complir aquest.

Com que de sistemes SCADA n'existexen varis en el mercat, també realitzem una comparació entre aquests per poder appendre a escollir quin dels diferents sistemes observats s'adapta millor a les nostres necessitats o al sistema que es vol crear.

També s'estudien els diferents elements de que es pot compondre un sistema automàtic, així com les comunicacions entre aquests elements. Elements com són el PC, el PLC, les tarjes d'adquisició de dades, els diferents softwares, ... Ja que sense ells seria impossible l'automatització d'una planta o sistema i amb la qual cosa no podríem realitzar un sistema SCADA d'aquest.

També s'estudia el disseny d'una interfície gràfica per PC amb un software específic com és LookOut, ja que és una realització bàsica per la creació d'un sistema SCADA, generant un exemple d'aplicació amb aquest programa en el control i supervisió i adquisició de dades d'una planta física com és la planta MPS1 de la maqueta de FESTO.

Paraules clau (màxim 10):

Monitorització	Interfície Gràfica	PLC	SCADA
Temps real	PC	Supervisió	HMI
Automatització	Control		

1. INTRODUCCIÓ.....	6
2. EN QUE CONSISTEIX UN SISTEMA SCADA.....	8
PRESTACIONS.	9
NECESSITATS BÀSIQUES DEL SISTEMA.....	9
MÒDULS D'UN SISTEMA SCADA.....	9
2.1.1. Configuració.	10
2.1.2. Interface gràfic de l'operador.	10
2.1.3. Mòdul de procés: resposta en temps real.	10
2.1.4. Gestió i arxiu de dades.....	10
2.1.5. Comunicacions.	10
DIFERÈNCIES ENTRE SISTEMES SCADA I DCS.....	11
3. HARDWARE EN SISTEMES DE SUPERVISIÓ.....	11
FUNCIONS DELS PC'S EN LA INDÚSTRIA.	12
ELS PC'S I ELS PLC'S.....	12
TARGES D'EXPANSIÓ.....	14
ESTRUCTURES OBERTES.....	14
4. SOFTWARE D'UN SISTEMA SCADA.....	15
SOFTWARE SCADA LOOKOUT.	15
4.1.1. Registres històrics de Lookout.....	15
4.1.2. Gràfics de Lookout.	15
4.1.3. Drivers inclosos a Lookout.	15
4.1.4. Seguretat.	16
4.1.5. Possibilitat de connexió en xarxa.....	16
SOFTWARE LABVIEW.	16
4.1.6. Que és LabVIEW.....	16
4.1.7. Objectius de LabView.	17
4.1.8. Panell frontal i diagrama de blocs.....	17

5.	COMUNICACIÓ DE SISTEMES SCADA.....	18
	ADQUISICIÓ DE DADES.	18
5.1.1.	Funcions generals de condicionament de senyal	19
5.1.2.	Consideracions generals sobre les TAD	19
5.1.3.	Diagrama de bloques general de una TDA	20
5.1.4.	Software de manejo de las TAD	20
	COMUNICACIONS	21
5.1.5.	L'estandard RS-232	21
5.1.6.	Sistema de bus i bus GPIB.....	22
5.1.7.	Norma IEE488.1 GIPB	23
5.1.7.1.	Norma IEEE 488.2 GPIB	24
5.1.8.	Xarxes d'àrea local LAN	24
5.1.9.	Internet i protocol TCP/IP	25
5.1.9.1.	TCP-IP en LabVIEW	26
6.	DIFERÈNCIES I COMPARACIONS ENTRE DIFERENTS SISTEMES SCADA.....	28
7.	APLICACIÓ.	29
7.1.1.	Pantalla principal del programa LookOut.....	30
7.1.2.	Creació d'una nova aplicació.....	30
7.1.3.	Introducció d'objectes fixos.....	32
7.1.4.	Inserció de parts mòvils o canviants.....	33
7.1.4.1.	Comandaments: interruptors i polsadors.	33
7.1.4.2.	Inserció de potencímetres.	35
7.1.4.3.	Inserció d'actuadors.....	36
7.1.4.4.	Indicadors.	38
7.1.4.5.	Inserció d'etiquetes.....	39
7.1.4.6.	Inserció d'alarmes.	39
7.1.4.7.	Posada en marxa de l'aplicació.	41
8.	CONCLUSIONS.....	43
9.	BIBLIOGRAFIA.	44

1. Introducció.

Donat un sistema automàtic de producció format per la part operativa o procés i la part de mando o controlador és necessari dotar a aquest sistema d'algun tipus de mecanisme que permeti la interacció de l'operador amb el sistema. A aquests tipus de sistemes se'ls anomena genèricament "interfície home màquina" (HMI) o sistemes de supervisió i explotació. Existeixen diversos tipus d'instruments a implementar, i un d'ells és els sistemes SCADA.

Avanç i encara avui, aquests sistemes de supervisió era implementat mitjançant panells formats per polsadors, leds de colors, potenciómetres i diplays de 7 segments. Els polsadors permetien a l'operador donar consignes de funcionament al sistema automatitzat, com per exemple, posar en marxa o parar el funcionament de la planta. L'operador, per mitjà dels leds observa si determinades parts dels automatismes estan operant o no. Els potenciómetres per la seva part permeten a l'operador donar referències de funcionament o fixar valors numèrics. I els diplays constitueixen la forma natural per mitjà del qual el sistema automatitzat trasllada a l'operador els marges de funcionament dels llaços de regulació.

Aquests tipus de panells proporcionen una pobre interfície respecte a les situacions d'alarma que simplement eren senyalitzades per mitjà de senyals acústiques i el parpadeig dels leds lluminosos, i a més no proporcionen cap tipus de mecanisme que permeti registrar els valors d'evolució del sistema automatitzat ni molt menys el seu anàlisi posterior.

Amb el desenvolupament dels sistemes basats en microprocessadors es van desenvolupar un tipus d'instruments programables anomenats "panells d'operador" que permeten portar a cap totes les operacions que poden realitzar-se amb les antigues botoneres, però amb un major nivell de flexibilitat ja que són programables i poden ser reutilitzats o modificats fàcilment si les condicions del procés varien. A més poden visualitzar a l'usuari gràfics que representen l'evolució del procés monitoritzat.

Amb aquests panells d'operador i els PC's, és quan es va començar a desenvolupar els sistemes anomenats SCADA.

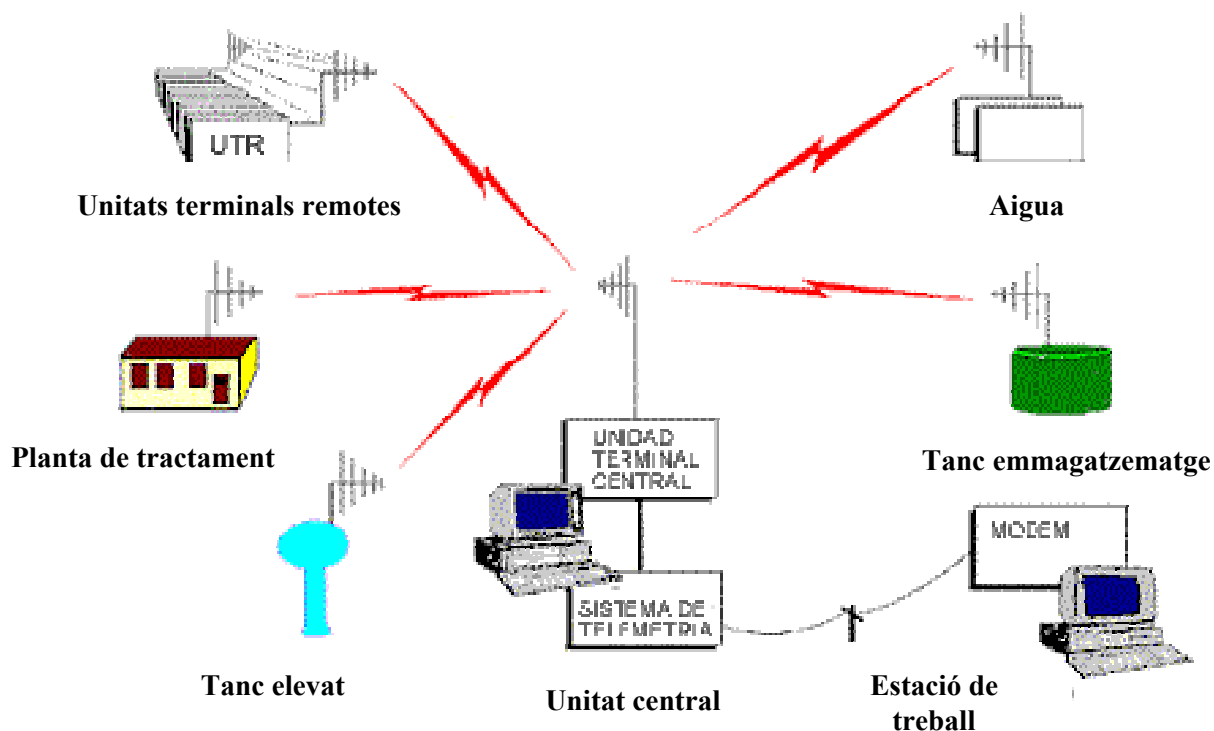
SCADA prové de les inicials de "Supervisory Control And Data Acquisition", i traduïnt-ho significa adquisició de dades i control de supervisió. Es tracta d'una aplicació software dissenyada especialment per funcionar sobre PC's industrials en el control de producció, proporcionant comunicació amb dispositius com controladors autònoms, autòmats programables... , i controlant el procés de forma automàtica des del monitor del PC. A més, ens indica tota la informació que es genera durant el procés productiu a diversos

usuaris, tant del mateix nivell, com d'altres supervisors dins de la empresa o cadena de producció, com poden ésser controls de qualitat, supervisió, manteniment, errors en el procés de producció, avaries, etc.

Dit d'una altra manera, aquest tipus de sistemes, normalment hi ha un PC que efectua tasques de supervisió i gestió d'alarmes, així com el tractament de dades i control de processos. I la comunicació que es realitza es fa mitjançant busos especials o reds LAN. Tot això s'executa normalment en *temps real*, i estan dissenyats especialment per donar a l'operador de la planta, la possibilitat de supervisar i controlar tots els processos que controlen aquests tipus de sistemes.



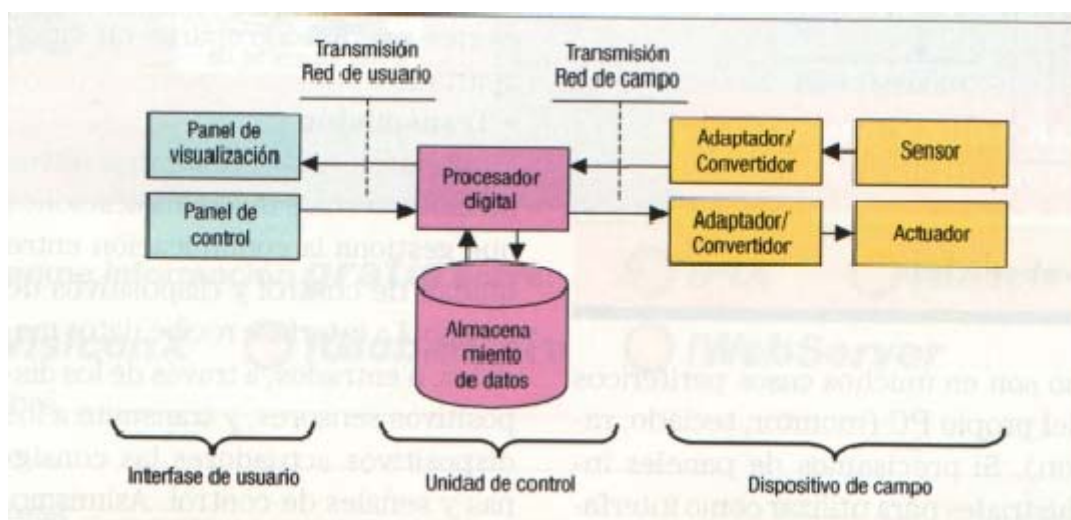
Tot això, juntament amb els programes necessaris i el hardware adicional que es necessita, es denomina sistema SCADA.



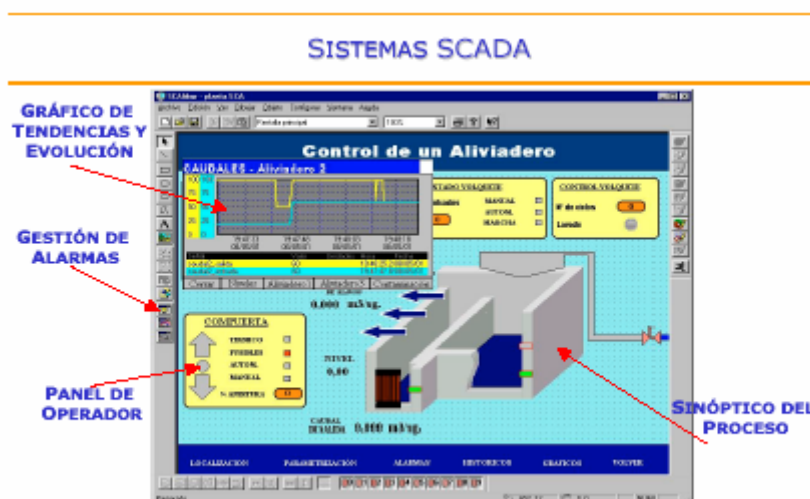
Són sistemes compostos bàsicament d'una o varies estacions centrals que són elements d'interfase Home-Màquina i que permeten a l'operador adquirir dades i realitzar controls sobre un conjunt de dispositius anomenats estacions distribuïdes o remotes. I tal com em dit anteriorment, es realitza una comunicació entre les estacions centrals i remotes mitjançant un protocol de comunicacions que pot ser alàmbic o inalàmbic, per obtenir una adquisició de dades de entrades/sortides de variables analògiques o digitals.

2. En que consisteix un sistema SCADA.

Els sistemes SCADA consisteixen primordialment de cinc components principals: un PC central, una terminal remota, perifèrics per a la comunicació, software de l'aplicació i transductors.



Aquests sistemes, poden treballar en un entorn on s'opera amb Windows, Unix, i altres sistemes de reds aplicats a sistemes d'aquesta naturalesa. A més, per al desenvolupament d'aplicacions SCADA disposem d'un software molt ampli com Factory Link, Fix DMACS, Wonderware Factory Suite, GE Simplicity, Citect, RSVIEW, Labtech, **LookOut**, **LabView**, BridgeView, DSC. A més, podem desenvolupar aquest software utilitzant eines com Visual Basic, C++, Delphi sota Windows NT o UNIX, si ho necessitem.



Prestacions.

Un paquet SCADA ofereix les prestacions següents:

- Possibilitat de crear panells d'alarma, que exigeixen la presència de l'operador per reconèixer una parada o situació d'alarma, amb un registre d'incidències.
- Generació d'històrics de senyal de planta, que poden ser bolcats per al seu procés sobre una fulla de càlcul.
- Execució de programes, que modifiquen la llei de control, o inclòs anular o modificar les tasques associades a l'autòmat, segons certes condicions.
- Possibilitat de programar numèricament, que ens permetrà realitzar càlculs aritmètics d'elevada resolució sobre la CPU del PC.

A més, alguns SCADA ofereixen llibreries de funcions per llenguatges generals que permeten personalitzar de manera molt ampla l'aplicació que es vol realitzar amb aquest SCADA.

Dit això, les funcions doncs que ha de complir un sistema SCADA són:

Necessitats bàsiques del sistema.

Un SCADA ha de complir varis objectius per que la seva instal·lació sigui perfectament aprofitable:

- Han de ser sistemes d'arquitectura oberta, capaços d'ampliar-se o adaptar-se segons les necessitats canviants del procés o de la planta en si.
- S'han de comunicar amb total facilitat i de forma transparent a l'usuari amb l'equip de planta i amb la resta de la planta (reds locals i de gestió).
- Han de ser programes sencills d'instal·lar, sense excessives exigències de hardware, i fàcils d'utilitzar, amb interfaces amigables amb l'usuari.

Mòduls d'un sistema SCADA.

Els mòduls o blocs de software que permeten les activitats d'adquisició, supervisió i control són els següents: configuració, interface gràfic de l'operador, mòdul de procés, gestió i arxiu de dades i comunicacions.

2.1.1. Configuració.

Permet a l'usuari definir l'entorn de treball del seu sistema SCADA, adaptant-lo a l'aplicació particular que es desitja desenvolupar.

2.1.2. Interface gràfic de l'operador.

Proporciona a l'operador les funcions de control i supervisió de la planta. El procés es representa mitjançant gràfics sinòptics emmagatzemats a l'ordenador de procés i generats des de l'editor incorporat al sistema SCADA o importats des d'una altra aplicació durant la configuració del paquet.

L'editor gràfic que incorporen permeten a l'enginyer la generació de les pantalles gràfiques de supervisió mitjançant sinòptics gràfics emmagatzemats en el PC de procés i generats des de l'editor incorporat en el sistema SCADA o importats des d'alguna altra aplicació durant la configuració del paquet.

2.1.3. Mòdul de procés: resposta en temps real.

Executa les accions de comandament preprogramades a partir dels valors actuals de les variables llegides. La programació es realitza mitjançant blocs de programa en llenguatge d'alt nivell (com C, BASIC,...).

Això requereix que el nostre sistema tingui la capacitat suficient i necessària en programes de processament de dades perquè sempre estigui a punt per processar i proporcionar els resultats dins d'un temps especificat, es a dir, que actui en temps real. Aquest context de *temps real* significa que un sistema reacciona als diferents events externs dins d'un temps especificats en un 100% dels casos, i respondre en temps concrets també en un 100% dels casos.

2.1.4. Gestió i arxiu de dades.

S'encarrega de l'emmagatzematge i processat ordenat de les dades, de forma que una altra aplicació o dispositiu pugui tenir accés a elles.

2.1.5. Comunicacions.

S'encarrega de la transferència d'informació entre la planta i l'arquitectura hardware que soporta el sistema SCADA, i entre aquesta i la resta d'elements informàtics de gestió.

Diferències entre sistemes SCADA i DCS.

ASPECTE	SCADAs	DCS
TIPUS D'ARQUITECTURA	CENTRALITZADA	DISTRIBUÏDA
TIPUS DE CONTROL PREDOMINANT	SUPERVISORI: Llaços de control tancats per a l'operador. Adicionalment: control seqüencial i regulatori.	REGULATORI: Llaços de control tancats automàticament per al sistema. Adicionalment: control seqüencial, batch, algorismes avançats, etc.
TIPUS DE VARIABLES	DESACOPLADES	ACOPLADES
ÀREA D'ACCIÓ	Àrees geogràficament distribuïdes.	Àrea de la planta.
UNITATS D'ADQUISICIÓ DE DADES I CONTROL	Remotes, PLCs.	Controladors de llaç, PLCs.
MITJANS DE COMUNICACIÓ	Radio, satèl·lit, línies telefòniques, connexió directa, LAN, WAN.	Xarxes d'àrea local, connexió directa.
BASE DE DADES	CENTRALITZADA	DISTRIBUÏDA

3. Hardware en sistemes de supervisió.

El hardware que podem escollir en sistemes de supervisió i control està entre escollir un PLC i un PC. En les tasques automatitzades de control, visualització i computació poden ser efectuades per PLCs (connectats en red mitjançant mòduls adequats) millor que amb sistemes exclusius de control basats en PC.

Així, per exemple, els actuals coneixements i preferències de l'usuari poden jugar un paper més gran que la pura potència de l'ordenador. Els factors crucials són els atributs de capacitat en temps real i les propietats de seguretat que fins ara han estat fortament associades amb el PLC, encara que el PC també pugui disposar de la característica de capacitat en temps real. Un sistema de control és inconcebible sense capacitat en temps real. Es comú en sistemes de control per ordenador haver d'elegir, segons les característiques dels sistema a supervisar, entre el PLC o el PC. I s'ha d'escollir el hardware que millor s'adapti a les necessitats del sistema a supervisar.

Els controladors lògics programables, en la majoria dels casos, estan dissenyats específicament per ser posats en ambients industrials exigents i han estat continuament desenvolupats de forma que els seus sistemes operatius en temps real representen la seva major virtut. Ells són i seguiran éssent la primera elecció per tot control de feines crítiques o extremes per al seu rendiment i simpleza, en els que un PC podria estar simplement "sobrecarregat" degut al treball que li pot suposar altres feines d'àmbit comú, com la gestió i visualització de dades, accesos a perifèrics, bases de dades, etc... Si a més del control de tares, es necessita un processament de dades, treball en red o visualització (una aplicació SCADA), un sistem basat en PC s'ha d'escollir amb consideració.

Funcions dels PC's en la Industria.

Els PC's tenen gran quantitat d'aplicacions en àrees molt diverses degut fonamentalment a que són dispositius programables de propòsit general molt eficients en el tractament de grans volums d'informació no crítica en el temps, i amb interfícies amb l'usuari molt intuïtives.

Referent a les tasques pròpies d'un sistema automàtic de producció, les tasques de control i regulació portades a terme en els nivells més baixos del model jeràrquic d'automatització manegen petits volums d'informació a gran velocitat en unes condicions de treball generalment adverses. A més, exigeixen unes altes condicions de seguretat i fiabilitat del sistema de control, per lo que tradicionalment els equips de control programables més utilitzats per resoldre-les ah estat els PLC's.

Des de fa poc tems, i donat que a un PC no era possible exigir-li aquestes característiques especials, els PC's han estat especialment utilitzats per cobrir les funcions assignades als nivells de supervisió de planta i de control de producció, fonamentalment per l'elevat volum de dades que en ells es manegen, els temps de resposta necessaris no crítics i la potent interfície gràfica amb l'usuari que ofereixen.

Tot i això, donat el gran desenvolupament tecnològic que vivim avui en dia es possible utilitzar un PC com a mecanisme per a implementar els sistemes de control necessaris, però per això és necessari assegurar-se de que compleix amb els requisits referents a la seguretat, fiabilitat, tems de resposta, ... A aquests equips de control se'ls anomena PC's industrials.

Avui en dia, l'estandarització arquitectònica que té un PC industrial, juntament amb els seu baix cost, la seva major potència de càlcul, la seva gran velocitat i sobre tot la seva major fiabilitat de funcionament i robustesa mecànica, està fent que el PC es contempli com una alternativa viable en els nivells més baixos de la jerarquia d'automatització. Tradicionalment i encara avui en moltes empreses, aquest nivell usa encara els sistemes PLC's o reguladors digitals.

Els PC's i els PLC's.

El PC industrial comença a ser considerat com un possible equip més en el control de processos. Això perquè si establim una comparativa entre les propietats dels PC's i dels PLC's es veu clarament que els PC's avantatgen amb gran diferència als PLC's en aquelles característiques que tradicionalment ho han fet, com són la potència de càlcul, la interfície amb l'usuari, la quantitat de memòria, etc. Tot i això, els que està sent determinant per la seva implantació en la indústria és la reducció de la distància que tradicionalment els ha separat en característiques com la robustesa, fiabilitat, etc.

En resum, els motius principals per lo que els PC's comencen a tenir presència com a equips de control són:

- Menor preu que els PLC's.
- Major capacitat de càlcul.
- Major disponibilitat de memòria.
- Major gamma de llenguatges de programació.
- Potents interfícies amb l'usuari.

Observem gràficament les diferències entre els dos dispositius en la següent taula:

PC's VS. PLC's		
		
HARDWARE		
---	ROBUSTEZ MECÁNICA	+++
---	SUSCEPTIBILIDAD ELECTROMAGNÉTICA	+++
+++	CAPACIDAD DE MEMORIA	---
-	CAPACIDAD DE AMPLIACIÓN DE E/S	++
SOFTWARE		
++	S. O. MULTITAREA	++
+++	SUBROUTINAS Y PRIORIZACIÓN	-
+++	MATEMÁTICAS EN COMA FLOTANTE	---
+++	LENGUAJES DE PROGRAMACIÓN	---
++	INTERFAZ	--
---	SEGURIDAD	+++

Pero cal evaluar si el PC davant altres equips de control programable és tan bo, perquè encara avui en dia es fan servir més els PLC's i no els PC's, per les següents raons:

- Pocs sistemes operatius en temps real.
- Pobra adaptació en el medi industrial (mecànica i elèctrica).
- Poques E/S suportades.
- Poca tolerància a fallos.
- Lenta reinicialització i entrada en funcionament.

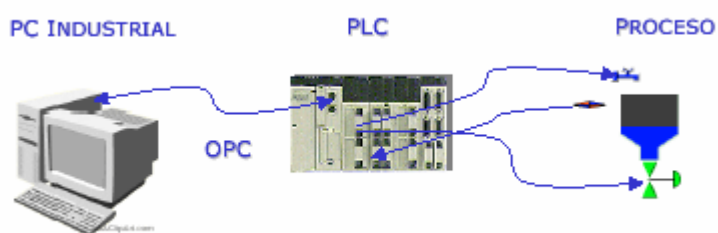
Dit això, veiem com queda estructurada la piràmide jeràrquica d'automatització, i en quins nivells doncs s'utilitzen uns equips o uns altres d'acord amb les seves característiques:



Si en aquesta piràmide observem en el seu interior, veiem quin tipus de sistemes corresponen a cada nivell:



Per això, els PC s'utilitzen en aplicacions de supervisió i adquisició de dades, interconnectats, això sí, amb els PLC's, que són els qui realment governen i controlen els processos:



Cal esmentar que s'usa el PC en la indústria ja que aquest, al gestionar el pas i la recepció d'informació entre els nivells superiors i inferiors en la jerarquia de control, és necessari un equip en que la seva capacitat de connexió amb altres components d'aquesta piràmide sigui el major possible i que asseguri la seva ràpida i fàcil integració, i que permeti la utilització de components estàndard tant hardware com software.

Targes d'expansió.

Com el sistema operatiu sols pot proporcionar respostes suaus en temps real, el més simple és emprar extensions hardware per les tareas crítiques i solucions software per la resta de tareas. Això ens porta a una compatibilitat en futurs sistemes operatius i una solució totalment factible actualment. Aquestes tarjetes d'expansió assumeixen les feines crítiques en temps real que el PC no pot atendre, estem parlant de tarjes que incorporen DSPs (Processadors de Senyals Digitals) o microcontroladors i que aporten una ajuda a l'anterior "sobrecarga" mencionada en l'apartat anterior.

Estructures obertes.

Encara no s'ha establert un estàndard per posseir extensions compatibles en temps real de sistemes operatius. De una forma estrictament determinant, els sistemes estàndard actuals han de ser modificats de forma general, així que la principal ventatja d'un sistema basat en PC pot arribar a ser un gran inconvenient. No obstant, l'estructura oberta, permet a l'empresa més llibertat en l'elecció de l'eina adequada per l'anàlisi, disseny i programació del sistema SCADA. Ara cada empresa ofereix la seva solució i la conversió a futures generacions de sistemes operatius ho fa cada vegada més difícil.

4. Software d'un sistema SCADA.

Software SCADA Lookout.

És un Software de supervisió i control que corre amb Windows 95/98/NT. Permet no solament el control d'un procés des d'un PC, sinó que també emmagatzema el registre històric de dades, i la carga de receptes de productes.

Lookout té 2 versions: Desenvolupament més Runtime, i solament Runtime. Totes dues permeten 500 punts d'entrades/sortides.

A més conté un excelent driver de comunicacions amb tota la línia PLC Direct, sinó a més conté 10 drivers addicionals per als PLC's i RTU's més utilitzats, i conta també amb la possibilitat d'intercanviar dades amb altres programes per DDE (Dynamic Data Exchange).

Lookout Direct no requereix, per a la seva configuració, de l'escriptura de programes ni de scripts. La seva arquitectura es orientada a objectes, i manipulada per events, lo qual redueix notablement el temps de resposta del sistema, davant a arquitectures manipulades per llaç.

4.1.1. Registres històrics de Lookout.

Es disposa de varis mecanismes per a aquests registres:

- 1) Base de dades encriptada. Pot analitzar-se mitjançant els gràfics del Lookout Direct, o bé llegits des d'un altre programa, per exemple Microsoft Access, mitjançant un driver ODBC.
- 2) Generació de taules de reportament històric, en formato .csv, llegibles, per exemple, des d'una taula o fulla de càlcul, com Microsoft Excel.
- 3) Reports d'events i alarmes, en arxius imprimibles.

4.1.2. Gràfics de Lookout.

Lookout combina gràfics de tendència en temps real, i gràfics històrics, en un únic gràfic. Quan s'estan analitzant valors històrics, es disposa d'una finestra amb la informació continguda a la base de dades històriques. A més, disposa d'un cursor que dona els valors precisos en las interseccions amb el gràfic, i amb la possibilitat de saltar a un punt específic en el temps. Disposa també de funcions de recorregut i de zoom.

4.1.3. Drivers inclosos a Lookout.

PLC Direct

Allen-Bradley PLC5

Allen-Bradley SLC500

ASCII

Delta-Tau (motion control)

GE-Series 90

Mitsubishi FX

Modicon Modbus ASCII, Plus, RTU

Omron

OPCClient

TIWAY

Si a més el seu equip amb el que es comunica ho permet, hi ha la possibilitat de comunicar-se per via telefònica, o per Ethernet.

4.1.4. Seguretat.

Poden crear-se usuaris amb diferents nivells d'accessos autoritzats. Permet prevenir l'accés a determinades funcions, mentres funcioni el sistema de seguretat, si es desitja, transparent a l'usuari.

4.1.5. Possibilitat de connexió en xarxa.

Permet la possibilitat de compartir informació a través de la xarxa amb altres màquines, inclús funcionant aquestes amb altres aplicacions.

Poden veure's pantalles simultàniament en diferents nodes, ajustar sets i reconeixer alarmes des de qualsevol node, configurar nodes per al monitoreig solament, etc.

Software LabVIEW.

4.1.6. Que és LabVIEW.

LabVIEW es una eina dissenyada especialment per a monitoritzar, controlar, automatitzar i realitzar càlculs complexes de senyals analògiques i digitals capturades a través de tarjetes d'adquisició de dades, ports sèrie i GPIBs (Bus d' Intercambi de Propòsit General).

És un llenguatge de programació de propòsit general, com és el Llenguatge C o Basic, però amb la característica que és totalment gràfic, facilitant d'aquesta forma l' entendiment i maneig d'aquest llenguatge pel disenyador i programador d'aplicacions tipus SCADA.

Inclou llibreries per a l'adquisició, anàlisis, presentació i emmagatzematge de dades, GPIB i ports sèrie. A més d'altres prestacions, com la connectivitat amb altres programes, per exemple de càlcul, i en especial MatLAB.

Està basat en la programació modular, lo que permet crear tarees molt complicades a partir de mòduls o sub-mòduls molt més sencills. A més aquests mòduls poden ésser usats en altres tarees, amb el qual permet una programació més ràpida i profitosa.

També ofereix la ventatja de "debugging" en qualsevol punt de l'aplicació. Permet la possibilitat de posar "break points", execució pas a pas, execució fins a un punt determinat i es pot observar com les dades van agafant i canviant valors a mesura que es va executant l'aplicació. A més també porta incorporat generadors de senyals per a poder realitzar un simulador.

4.1.7. Objectius de LabView.

LabVIEW és un llenguatge completament gràfic, i el resultat d'això és que és totalment semblant a un instrument, per això a tots els mòduls creats amb LabVIEW se'ls denomina VI (Instrument Virtual).

Existeixen dos conceptes bàsics en LabVIEW: el Front Panel (Panell Frontal) i el Block diagram (Diagrama de Blocs). El Panell Frontal és la interfície que l'usuari està veient i pot ésser totalment semblant a l'instrument del qual s'estan recollint les dades, d'aquesta manera l'usuari sap de manera precisa quin és l'estat actual d'aquest instrument i els valors de les senyals que s'estan mesurant, El diagrama de blocs és el conexasionat de tots els controls i variables, que tindria certa semblança al diagrama de l'esquema elèctric de l'instrument.

LabVIEW té la característica de descomposició modular ja que qualsevol VI que s'està dissenyant pot convertir-se fàcilment en un mòdul que pot ser utilitzat com una sub-unitat dins d'un altre VI. Aquesta peculiaritat podria comparar-se amb la característica de procediment en els llenguatges de programació estructurada.

És un sistema obert, en quant a que qualsevol fabricant de tarjetes d'adquisició de dades o instruments en general pot proporcionar el driver del seu producte en forma de VI dins de l'entorn de LabVIEW. També es possible programar mòduls per a LabVIEW en llenguatges com C i C++, aquests mòduls són coneguts com Sub-VIs i no es difereixen als VI creats amb LabVIEW tret de la interfície del llenguatge en el que han estat programats. A més aquests Sub-VIs són molt útils per exemple en el camp de càlculs numèrics complexes que no es troben incluits en les llibreries de LabVIEW.

4.1.8. Panell frontal i diagrama de blocs.

Es podria dir que en qualsevol VI existeixen dos cares ben diferenciades: El Panell Frontal i el Diagrama de Blocs.

El Panell Frontal és la cara que l'usuari del sistema està veient quan s'està monitoritzant o controlant el sistema, o sigui, la interfície de l'usuari. Aquest conté controls i indicadors i existeix una gran varietat d'ells, però a més inclús es poden dissenyar controls i indicadors personalitzats, el qual permet tindre una àmplia gamma de dits controls i indicadors.

Un control pot adoptar moltes formes, i moltes d'aquestes formes el dibuix real usat en instruments reals. Altres són estrictament conceptes digitals o analògics. Però tots els controls tenen una forma visual que indiquen a l'usuari quin és l'estat d'aquest control a l'instrument real. És molt important en un sistema SCADA que l'usuari no tingui que interpretar res, sinó que tot sigui clar, ja que les interpretacions poden donar lloc a falses actuacions i, a causa d'això, podrien existir lamentables errors. A més, dos usuaris podrien interpretar de forma diferent qualsevol event.

El Diagrama de Blocs de l'VI seria la cara oculta del Panell Frontal, una cara que l'usuari del sistema no pot veure. En ella estan tots els controls i indicadors interconectats, assemblant-se molt a un diagrama d'un esquema elèctric. Aquesta cara es molt menys conceptual que el Panell Frontal i per a l'usuari seria molt difícil entendre-la.

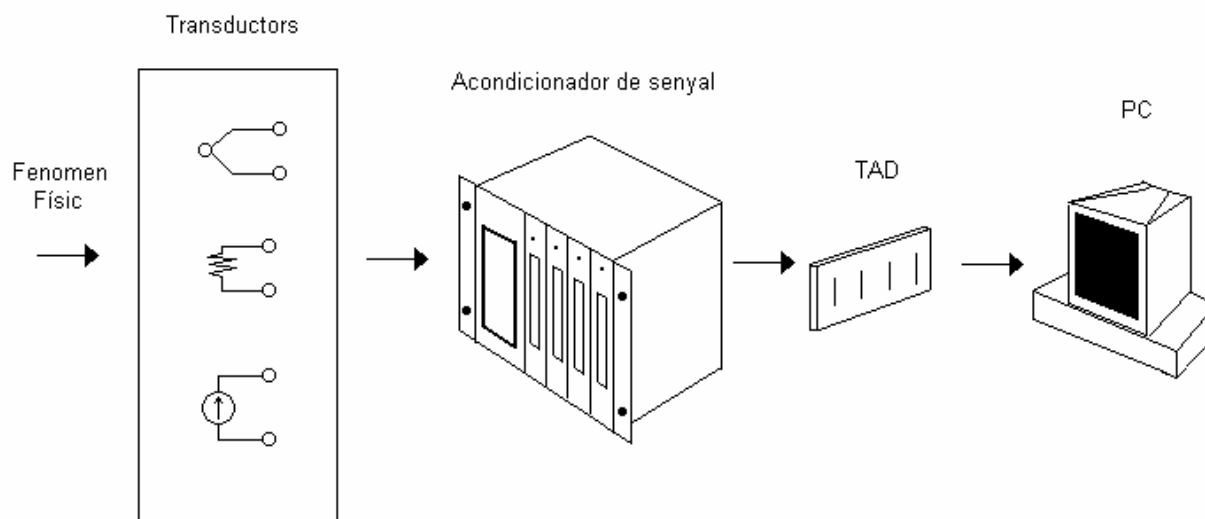
Tots els mòduls estan interconectats, mitjançant línies de connexió, per on circulen les diferents dades o valors de l'VI, d'aquesta manera s'aconsegueix que l'VI funcioni com un conjunt d'elements, mòduls i sub-mòduls.

5. Comunicació de sistemes SCADA.

Adquisició de dades.

En els sistemes SCADA la adquisició de dades és fa indispensable per el tractament de senyals que ens proporcionen informació sobre els fenòmens físics. En general aquest tractament es necessari fer-lo sobre grans quantitats d'informació i una elevada velocitat de processat; un ordinador personal és l'encarregat de realitzar aquestes feines degut a la seva gran velocitat de processat. Generalment, els dispositius utilitzats per la adquisició de senyals són les targetes de adquisició de dades, que son les que proporcionen a l'ordinador la capacitat de adquirir i generar senyals analògiques o digitals. Però, aquestes no són les úniques funcions que realitzen, també disposen de comptadors i temporitzadors.

Quan es desitja obtenir informació sobre fenòmens físics es necessari introduir un nou element en el sistema que ens subministri un paràmetre elèctric a partir de un paràmetre físic, aquest element es el transductor. El transductor es el primer element que forma un sistema general d'adquisició de senyals. Generalment, les senyals elèctriques generades per els transductors no son adequades o no son compatible amb les característiques de entrada de una tarja de adquisició de dades. En aquests casos es fa necessari l'ús de dispositius de condicionament de senyal que realitzen un pre-tractament de la senyal. Les funcions més usuals dels acondicionadors son amplificació, filtrat, aïllament elèctric, també liberalització i multiplexació.



Configuració general d'un sistema PC basat en l'adquisició de dades

5.1.1. Funcions generals de condicionament de senyal

Depenen del tipus de transductors que s'utilitzin, l'ús de un equip de condicionament de senyal pot millorar la qualitat i les prestacions del sistema d'adquisició. Les funcions de condicionament que s'utilitzen generalment són:

Amplificació. Degut al baix nivell de senyal que subministren els transductors, el soroll pot jugar un paper important pel que fa l'error de mida de senyal. Una ampliació fora del chasis del PC i propera a la font de origen de la senyal ens pot incrementar la resolució de la mida i reduir de una forma efectiva l'efecte del soroll sobre la senyal desitjada.

Filtrat. L'ús de filtres permet rebutjar un cert marge de freqüències indesitjables. És molt comú l'ús de filtres banda-eliminada amb freqüència central 50 Hz per eliminar el soroll de xarxa procedent de fluorescents, maquinaria, fonts d'alimentació, etc. També són comuns els filtres aliasing que permeten que la senyal que serà mostreigada pugi ser reconstruïda perfectament després de l'adquisició. L'ample de banda de arquets filtres té que coincidir amb l'ample de banda màxim de la senyal desitjada.

Aïllament. La incompatibilitat de masses entre les TAD i les senyals a medi és la causa més comú dels problemes de mida i poden arribar a fer mal bé la TAD. El mètode més utilitzat per l'aïllament consisteix en la utilització de circuits òptics.

5.1.2. Consideracions generals sobre les TAD

Es important conèixer quals són les prestacions que ens poden donar cada tarja per que s'adapti correctament a la nostre aplicació. Les consideracions que determinaran les característiques hardware de les targetes d'adquisició de dades, per tenir un criteri de valoració de la efectivitat de la TAD i de comparació entre diferents plaques.

Entrades Analògiques. Les prestacions i precisions que ens proporciona una tarja, per les entrades són bàsicament el número de canals de que disposa, la freqüència de mostreig, la resolució i els nivells d'entrada. Molts d'aquests paràmetres es poden configurar per software.

El número de canals analògics s'ha d'especificar tant per les entrades referenciades a massa com per les diferencials (single-ended inputs). Si entre els terminals de referència i terra existeix una diferència de potencial, aquesta es denominada tensió en mode comú, causant de molts errors de mida. Configuració en adquisició de senyals de alt nivell on l'error en mode comú és despreciable.

Les senyals diferencials es basen en que els dos diferencials d'una entrada corresponen amb dos terminals d'entrada de la TAD, sense terminal referenciat a massa. D'aquesta forma eliminem la tensió en mode comú. Aquesta configuració d'entrada és útil per l'adquisició de senyals de baix nivell.

Freqüència de mostreig. Determina la velocitat a la que es produeixen les conversions ADC. Una freqüència de mostreig elevada proporcionaria de senyals amb major qualitat de definició en el temps; però fa augmentar el flux de dades cap el processador. Per això, cal buscar un valor de compromís que faci òptim el funcionament del sistema. És fundamental en tota adquisició de dades respectar el teorema del mostreig de Nyquist.

Resolució. Indica el número de bits que utilitza el conversor ADC per quantificar els nivells de senyal analògica. Quan major sigui el número de bits del ADC, major sigui el número de nivells de senyal que es poden representar.

Nivells d'entrada. Són els límits de entrada de tensió de la TAD. Es molt comú diferenciar entre les senyals unipolars i bipolars. Les senyals unipolars admeten únicament nivells de tensió positius mentres que les bipolars permeten les dos polaritats.

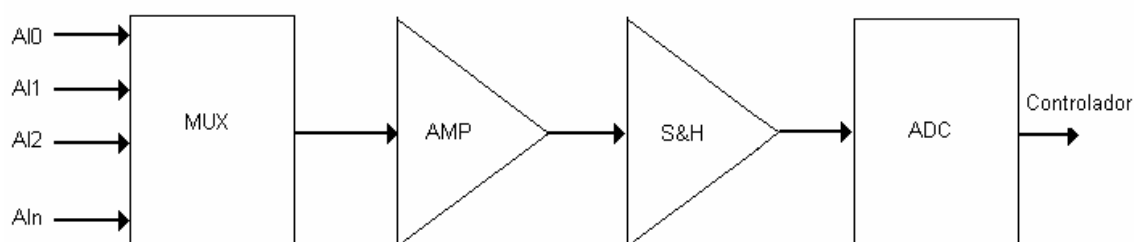
Sortides analògiques. Moltes TAD incorporen sortides analògiques. Bàsicament, les característiques tècniques de les sortides analògiques son les mateixes comentades per les entrades.

Ports digitals. Son línies d'entrada/sortida digitals. S'utilitzen pel control de processos, generació de models per testeig, per comunicació amb equips perifèrics,ect. Els paràmetres més importants que caracteritzen els ports digitals són el número de línies disponibles, la velocitat a la qual es poden transferir les dades i la capacitat de control de diferents dispositius.

Temporitzadors. Son línies útils per moltes aplicacions tals com comptar les vegades que es produeix un event, generar bases de dades de temps per processos digitals o generació de polsos.

5.1.3. Diagrama de bloques general de una TDA

La etapa d'entrada d'una TDA es molt comú pera tots els tipus t models. Bàsicament és composta per un multiplexor, que permet disposar de varis canals d'entrada, seguit d'un amplificador d'instrumentació de ganancia programable. Aquest amplificador es connecta a altre amplificador de mostreig i retenció ("Sample & Hold") i finalment aquest proporciona el valor de tensió el conversor ADC.



Etapa d'entrada general de una TAD

En Quant a les sortides analògiques es componen bàsicament de conversors DAC que connecten directament al bus intern del microprocesador. Per cada sortida analògica es necessita un conversor DAC que normalment tenen la mateixa resolució que els ADC de la entrada.

5.1.4. Software de manejo de las TAD

Tota placa d'adquisició de dades necessita un software de control. Aquest control es pot fer de tres formes diferents. La primera consisteix en la programació directa dels registres. Es el mètode que permet més flexibilitat per la capacitat de control, però també es el més costos en quant temps i dificultat de programació. Altre mètode es utilitzar un driver o software de control de la tarja. Es tracta d'una sèrie de funcions que actuen sobre els registres de la targeta, però a un nivell de programació superior al de la programació directa de registres. Proporciona la mateixa flexibilitat de programació que aquesta, però el temps d'aplicació disminueix de forma apreciable. Per últim, és pot controlar la tarja també a través d'un programa de nivell superior als anteriors esmentats. Això permet crear aplicacions potents per representació i anàlisis. Un exemple d'aquest mètode seria el control de la tarja a través de LabVIEW.

Comunicacions

Després de obtenir la senyal del traductor i condicionar-la debidament per que sigui compatible amb la tarja d'adquisició utilitzada, es necessari un camí per transmetre les dades preses en tots els dispositius a controlar i monitoritzar fins l'ordinador central. Per aquesta feina hi ha diferents sistemes de comunicacions però esmentarem els més utilitzats per sistemes scada.

5.1.5. L'estendard RS-232

Normalment, els dispositius que intervenen en una comunicació serie són *l'Equip Terminal de Dades* (ETD), que sol ser un PC, i *l'Equip de Comunicació de Dades*(ECD), generalment un modem. Però la gran implantació dels PCs ha deriva en l'ampliació del ús RS 232, convertint-se en el estàndar més utilitzat en aplicacions de baix cost que requereixen la interconnexió sèrie entre un ETD i un perifèric.

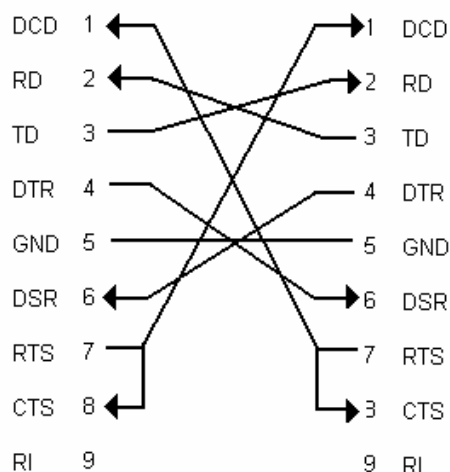
Les característiques principals són:

- Les senyals han de ser binàries i sense balancejar.
- La tensió no pot superar el 25 volts en circuit obert.
- La tensió d'utilització del equip pot ser positiva 0 lògic o negativa 1 lògic i el seu valor entre 5 i 10 volts.
- En cortcircuit la intensitat no pot superar 0,5 ampers.
- La resistència de carrega deu ser superior a 1000 ohms i no pot passar de 7.000 ohms.
- La capacitat de carrega màxima 2500pF. Això provoca limitar a una longitud màxima de cable entre PC i perifèric de 15 a 20 metres.
- Velocitat màxima de transmissió de dades: 20 Kb/s. Però, existeixen aplicacions que es surten de les especificacions del estàndards que arriben als 116kb/s.

El connector DB9S té 9 patilles es molt utilitzat en les connexions RS 232.

patilla	Sigles	Descripció
1	DCD	Data Carrier Detect
2	DSR	Data Set Ready
3	RD	Receive Data Line
4	RTS	Resquest To Send
5	TD	Transmit Data Line
6	CTS	Clear To Send
7	DTR	Data Terminal Ready
8	RI	Ring Indicator
9	GND	Signal Ground

En moltes ocasions la connexió sèrie no es realitza entre PC i modem, sinó entre PC i altre dispositiu com altre PC, un Autòmat Programable, o altres. En aquest cas es necessari construir un cable de comunicacions especial, “**modem nul**”.



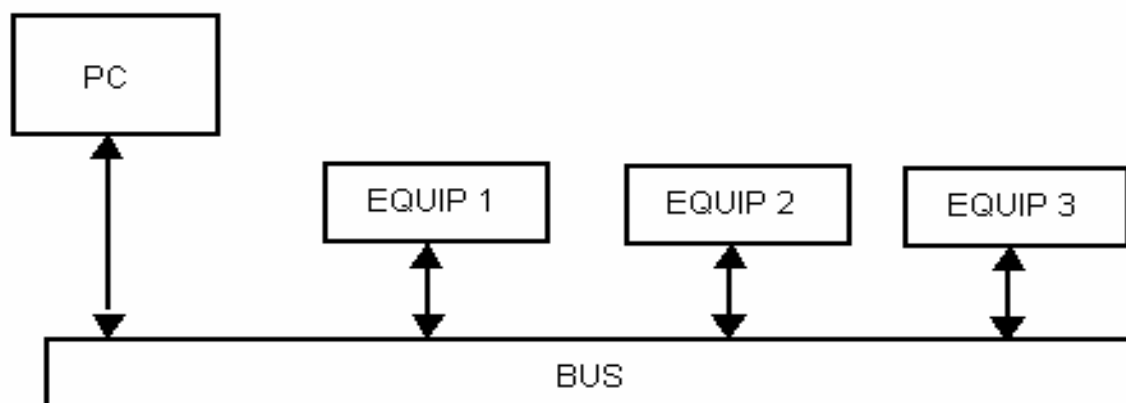
LabVIEW proporciona eines de gran utilitat per fer servir el port sèrie. Totes les funcions que són necessàries a l'hora de realitzar una comunicació sèrie entre el PC i un perifèric es troben lla programades en forma d'instruments virtuals.

5.1.6. Sistema de bus i bus GPIB

Un bus es un sistema intel·ligent on, seguint unes normes adequades de coordinació o protocol en el intercanvi de dades fa la interconnexió de diferents perifèrics per realitzar un procés concret.

El bus GPIB es un del més utilitzat dens del mon de la instrumentació virtual per interconnectar instruments de laboratori entre sí o amb els seus corresponents instruments virtuals, soportats sobre una plataforma hardware com pot ser un PC.

Hewlett-Packard va desenvolupa el GPIB original, anomenat HP-IB, a finals de la dècada dels setanta, quan va aparèixer els primers instruments controlables digitalment, llavors va sorgir la necessitat d'estandarditzar la comunicació entre ell i l'ordinador. En l'any 1975, IEEE va publica l'estàndard NSI/IEE 488-1975 norma a seguir per aquells fabricants **que desitjaven** utilitzar el bus GPIB i on s'indicaven quals eren les característiques elèctriques, mecàniques i funcionals del sistema de interface.



Com que aquesta primera norma no incloïa indicacions sobre la sintaxis o el format dels comandos a utilitzar va aparèixer la norma 488.2 que incloïa un mínim conjunt de missatges que deuria entendre l'instrument o l'equip de formats de dades o comandes.

5.1.7. Norma IEE488.1 GIPB

El bus transportarà informació en totes direccions. Tots els instruments rebran les mateixes dades i deuran reconèixer automàticament a qui ha sigut direccionades.

Els objectius de la norma IEEE488.1 eren:

- Crea un sistema d'instrumentació on els instruments estiguin pròxims.
- Especificar els requisits mecànics, elèctrics i funcionals per poder interconnexionar instruments.
- Permetre la comunicació directa entre instruments sense necessitat que les dades passin per el controlador.
- Definir un sistema que permeti interconexionar instruments de velocitats diferents, fabricants diferents i característiques o capacitats diferents.
- Definir un sistema que no introdueixi restriccions sobre les característiques dels instruments.
- Definir un sistema fàcil d'usar, de relatiu baix cost i que permeti comunicacions asíncrones en un ampli marge de velocitats de transmissió de dades.

Els paràmetres més importants d'aquest bus:

- Dades mecàniques i elèctriques: Tipus de connector o tecnologia a utilitzar.
- Bus orientat a cert processador o independència d'ell.
- Espai de memòria direccionable o quantitat d'instruments a direccionar.
- Ample de dades.
- Tipus de transferència de dades: Sincronia o asíncrona.
- Existència de multiplexació de dades i direccions.
- Freqüència de rellotge.
- Velocitat màxima de transferència de dades.
- Número d'interrupcions i protocol.
- Número d'unitats masters.

5.1.7.1. Norma IEEE 488.2 GPIB

En 1987 es va fer una revisió de la norma en la seva versió 1. Aquest pas va suposar un pas més en la compatibilitat d'instruments, introduint solucions com:

- Definir el mínim conjunt de capacitats que he de tenir un instrument.
- Especificar la forma de presentar les dades a través del bus.
- Es va definir un protocol per enviar missatges de dispositiu i la forma d'enviar varis missatges en una sola cadena de caràcters.
- Es va donar un conjunt de comandes comunes a tots els instruments.
- Es va definir el model estandar de bytes d'informació sobre l'estat del dispositiu.

5.1.8. Xarxes d'àrea local LAN

Una definició completa i actual de Xarxa d'àrea local seria: Sistema de comunicacions capaç de facilitar l'intercanvi de dades informàtiques, veu, facsímil, vídeo conferències, difusió de vídeo, telemetria i qualsevol altre forma de comunicació electrònica.

Existeix una definició oficial, del comitè IEEE 802, que defineix una Xarxa local de la següent forma: Una xarxa d'àrea local es un sistema de comunicació que permet que un número de dispositius independents es comuniquin entre si.

Una Xarxa local, com el seu nom indica, té que ser local en quant al àmbit geogràfic, això avarca desde un laboratori petit, una oficina, un edifici o un complex industrial amb dotzenes de naus.

Les principals característiques de les LAN es podrien resumir:

- Distàncies cobertes de 200m fins 5 Km.
- Us d'un medi de comunicació privat.
- Velocitats d'accés elevades de 0,2 a 16 Mb/s.
- Simplicitat del medi de transmissió que utilitza (cable coaxial, fibra òptica, cables telefònica).
- Gran varietat de dispositius connectats
- Admiteix qualsevol tipus de topologia.
- Possibilitat de connexió amb altres xarxes.

Els elements que configuren una xarxa son:

- servidor: Es el ordinador principal de la xarxa que suporta el sistema gestor de la mateixa, proporcionen tots el serveis a els terminals connectats a la xarxa.
- Estacions de treball: Componen els equips terminals de la xarxa, poden ser tan sols els terminals o bé PC's, amb els seus dispositius físics.
- Targes de xarxa: Són l'element interfaz que permet la connexió d'un ordinador a la xarxa, constitueixen el pilar sota el que es sosté tota la xarxa.
- El medi: constituït pel cablejat els connectors que enllacen els components de la xarxa.
- Dispositius addicionals: Són tots aquells components de la xarxa que són utilitzats i compartits per els usuaris de la xarxa; són recursos com: impressores, unitats de emmagatzematge.

5.1.9. Internet i protocol TCP/IP

TCP/IP és una eina de compartició i distribució de dades a través d'una xarxa d'àrea local o internet. La gran majoria de les aplicacions actuals recullen, analitzen, processen i visualitzen les dades en una mateixa màquina. Aquesta ideologia actualment lla esta canviant, ara la adquisició de dades lla no es realitza únicament del el PS, sinó que existeixen sensors distribuïts que recullen informació i es connecten directament a la xarxa athenet convertint-se en un punt de mida remot al qual anirem a connectar-nos i importarem les dades a la nostre aplicació.

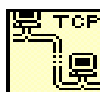
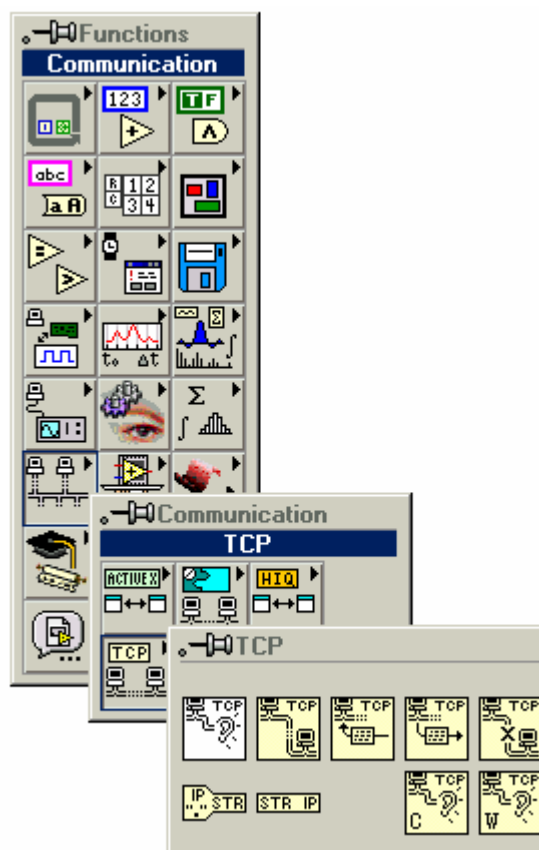
Altre de les ventatges de muntar aplicacions distribuïdes es el fet de processar informació en una màquina més potent i poder visualitzar els resultats en altre amb característiques més limitades.

Les eines que ens permeten comunicar-nos a sistemes independentment de la seva localització, son els protocols. Amb TCP-IP (Transmission Control Protocol) podem interconnectar-nos amb equips que es troben dintre i fora de la nostre xarxa local. Volem dir a qualsevol **equip** o **PC** que tingui una direcció IP, una vegada connectat amb aquest equip podrem transmetre i rebre dades.

D'aquesta manera podem connectar qualsevol dels nostres instruments a través de una interface TCP-IP a internet. Això obre moltes portes als sistemes remots ja que fins ara podíem controlar un instrument remot a pocs metres de distancia, però ara podem controlar i obtenir informació de ells en qualsevol part del planeta sense necessitat de crear una línia de transmissió perquè aquesta lla esta creada.

5.1.9.1. TCP-IP en LabVIEW

LabVIEW disposa d'una sèrie de instruments virtuals (VI) que implementen el protocol TCP-IP.



Obre una connexió TCP



Tanca una connexió TCP



Realitza una escritura de dades a la xarxa



Crea una llista per una connexió TCP



Realitza una lectura de dades procedent de la xarxa



Espera una connexió TCP



Crea una llista i espera una connexió TCP



Converteix un string en una direcció IP

Amb les funcions de LabVIEW podríem crear un servidor de dades, un exemple dels passos a seguir per crear un serien:

1. Escoltar i esperar a que un client demani una connexió.
2. Una vegada ens demanen connexió, se envien les dades de la següent manera:

Primer s'envia la longitud de trama de dades.

Segon se envia la trama de dades.
3. Una vegada s'han enviat les dades es tanca la connexió.

Un exemple dels passos a seguir per crear una client de dades.

1. Obrir la connexió en un port i en una direcció del servidor de dades.
2. Fer una primera lectura de 4 bytes on es llegirà la longitud de la trama de dades,
3. Realitzar una segona lectura. Ara de la trama de dades i llegirem tants bytes com indiqui la primera lectura.
4. Una vegada s'ha finalitzat la lectura de dades es tanca la connexió.

6. Diferències i comparacions entre diferents sistemes SCADA.

Com s'ha vist en punts anteriors els sistemes scada s'han convertit en aplicacions de supervisió, gestió, control de planta i de procés, són el nucli que proporciona informació en temps real als sistemes automatitzats de gestió integrada de informació en l'empresa, superant qualsevol limitació física i geogràfica.

La taula correspon a la versió base del software scada. En aquesta taula junt amb el nombre del producte, del fabricant i del distribuïdor s'indica per quines plataformes hardware hi ha versions disponibles del producte, també per que tipus de productes s'ha dissenyat, l'entorn operatiu en el que es pot executar i l'any de la primera i l'última versió. També es mencionen els principals mòduls que ofereixen els fabricants de software scada, els seus preus i característiques més significatives.

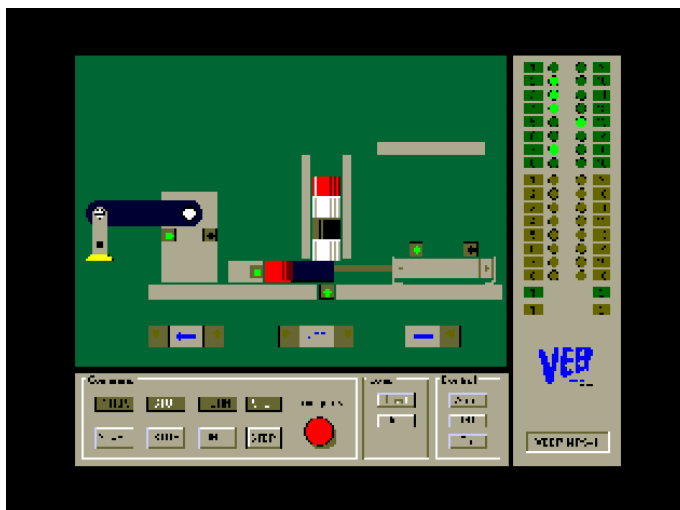
Nom del Paquet	Aimax	All-Data Automation	CX-Supervisor	Law View DSC	Lookout	Satt Graph 5000	Scada-V5	Simatic WinCC
Subministrador / Fabricant	Desing Instruments S/A T.A. Engineering U.S.A. desing@desing.com	Freixas y Ros S.L. frall-done@terra.es	Omron omron@omron.es	National Instruments nispain.general@ni.com	National Instruments nispain.general@ni.com	ABB www.abb.com	Foxboro (Grupo Foxcada Invents) (Australia) www.foxboro.com	SiemensPS eleon@ssa.siemens.es
Plataforma hardware	PC	PC	PC	PC, PXI	PC, PXI	PC	Indep.	PC
Tipus d'aplicacions	P,I,S/C	P	P	P,I	P,I	P	P	P,I
Sistemes operatius	W9x,NT,2K,ME,XP	W9x,MENT/2000/XP	W95/98/NT,Wme,W2000	Windows	Windows	WNT	Unix, WNT	WNT, ME W2000, 98SE
Fecha 1ª/Última versió	1985/02	2001/02	1996/01	1997/02	1997/02	1996/00	1995/99	1996/02
Drivers per buses	Profibus, FielbusF, Ethernet, Modbus I	De diversos PLCs	Omron	OPC	OPC/PLC	Applicom, Comili, DDE, Modbus, OPC, SATT BUS, Inteenal	Fieldbus Foundation, Profibus, Fox, Hart, Etchelon y otros	Profibus, Ethernet i altres fabric. De PLC i OPC
Driver OPC	C/S	-	C	C/S	C/S	C	-	C/S
Control d'usuaris	Si	NO	Si	Si	Si	Si	-	Si
Max. Freq de mostreig d'una única senyal	Segons Hardware	Segon PLC	Segons Bus	-	-	1Khz	-	>1Khz
Llenguatges de programació	Visual Basic y C++	Visual Basic 5.0,6.0, .NET	Scripts propis Java Scripts i Visual Basic	Gràfic	Gràfic	Scripts propis (molt semblant a ladder logic, seqüencial Pascal)	Gràfica, C++,blocs ladder logic, seqüencial	ANSI C
Integració amb altres aplicacions	Bases de dades, fulles de càlcul	MS Office	CX-Server, CX-Server OPC, CX-Server Lite i Programer	Qualsevol de Windows	Qualsevol de Windows	MS Visual, Basic, Excel, Acces, Word	SAP i altres	Simatic Step 7, aplicacions windows i externes
Tipus d'enllaç	OLE,DDE, ODBC	Active X	OLE, COM/DCOM, OPC,Activ X	OPC, DDE, Active X, DataSocket, TCP/IP	OPC, DDE, Active X	OLE, MMS, Gataway	ODBC direct Core TCP/IP	OPC, DDE,ODBC,DNA,COM/DCOM,OLE y Active X
Generació d'informes	Propi	Propi	Propi	Si	Si	-	Her. estandards	Propio
Funcionalitat d'informes	Definible	MS Acces	Formats ASCII, HTML, RTF, Excell i Access	-	-	-	Per demanda per l'operador	Personalitzables per usuari. Sist. Modular
Preu mínim versió desenvolupament	Consultar	1244 Euros	1.625 run-time 835	2700 Euros run-time 750 Euros	1.000 Euros run-time 540 Euros	Consultar	18000 Euros	Consultar
Configuració màx. d'aquesta versió mín.	32.000 tags	Sense límit	-	limitada	limitada	Consultar	limitada	128 variables externes.Var.Inter sin límite
Preu desenvolupament versió completa	Consultar	2074Euros	1.625 Euros run-time 835Euros	-	6490 Euros run-time 4.400 Euros	-	Consultar	Consultar

7. Aplicació.

L'objectiu del treball és (ja comentat de que es compona un sistema SCADA), fer un aplicatiu amb un d'aquests sistemes SCADA.

El sistema SCADA que nosaltres em escollit és LookOut de la casa National Instruments. I amb aquest controlarem la maqueta de FESTO, en concret la planta MPS1.

Per poder realitzar l'SCADA de la MPS1, em extret una imatge del programa VEEP, on ja tenim dibuixats tots els components de que es composarà el nostre aplicatiu.



I sobre aquesta imatge anterior s'haniran extreient tots els elements que necessitem o treurem tots els que no ens faran falta.

En la MPS1, els elements a controlar són (veure fig.1):

1. El cilindre expulsor de peces.
2. El manipulador.
3. La ventosa del manipulador que agafa les peces.
4. L'alimentador.

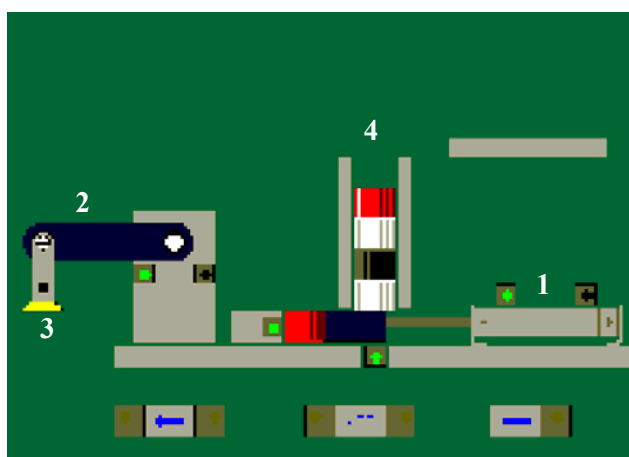
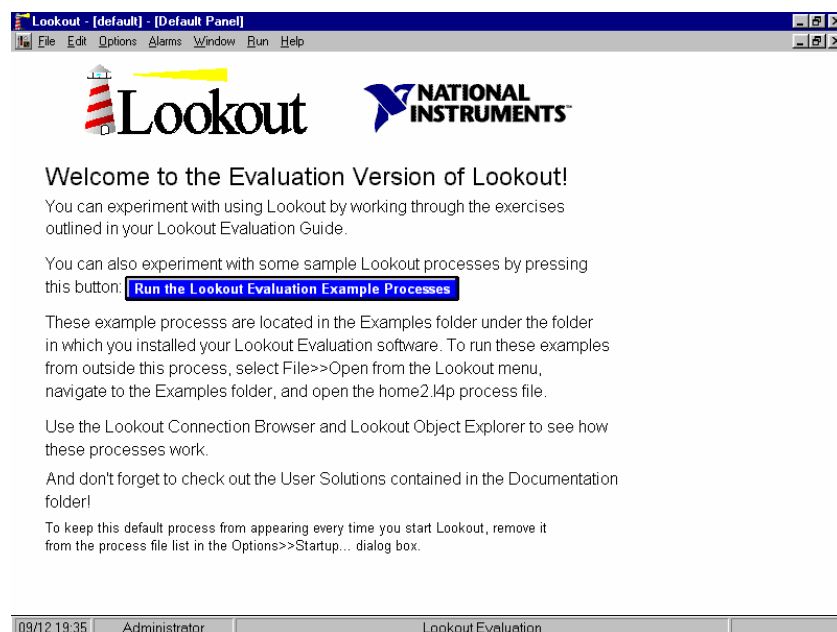


Fig. 1

Els elements a visualitzar són les peces, que d'acord amb els diferents senyals de sensors i actuadors em d'anar visualitzant els diferents estats en que es troben les peces.

7.1.1. Pantalla principal del programa LookOut.

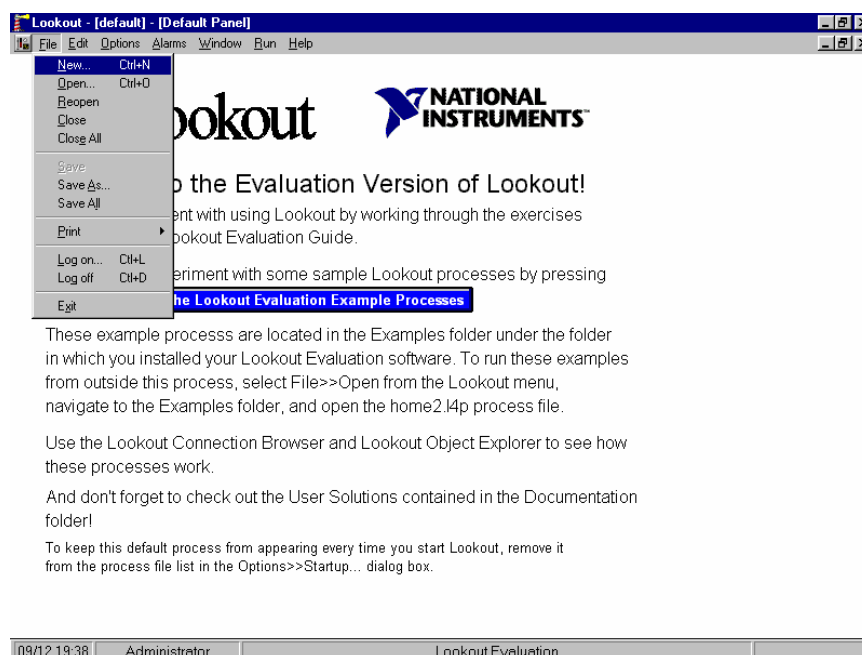
Quan inicialitzem el programa ens apareix una pantalla com aquesta:



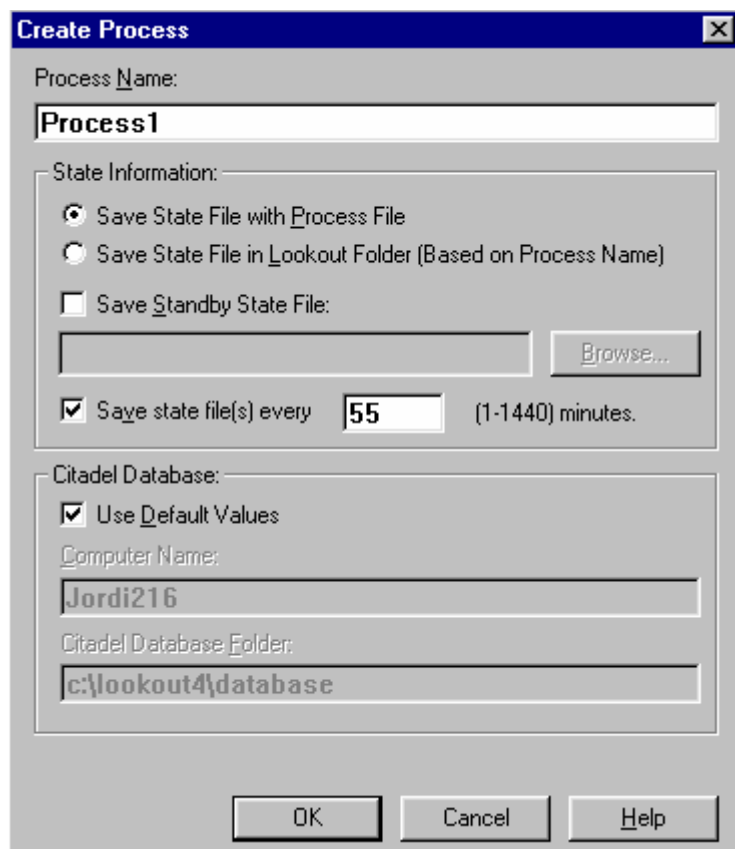
Si volem visualitzar qualsevol dels exemples que ja porta instal·lats el programa, clicarem sobre la pestanya que ens apareix en color blau.

7.1.2. Creació d'una nova aplicació.

En el nostre cas, es crear una nova aplicació. Per lo tant em d'anar a al menú en la pestanya **file/new**:



Ens apareixerà una nova finestra en la que haurem de posar com s'anomenarà el procés que es crearà i diverses opcions de grabació d'aquest:



Create Process

Process Name:

State Information:

- ☒ Save State File with Process File
- ☐ Save State File in Lookout Folder (Based on Process Name)
- ☐ Save Standby State File:

☒ Save state file(s) every (1-1440) minutes.

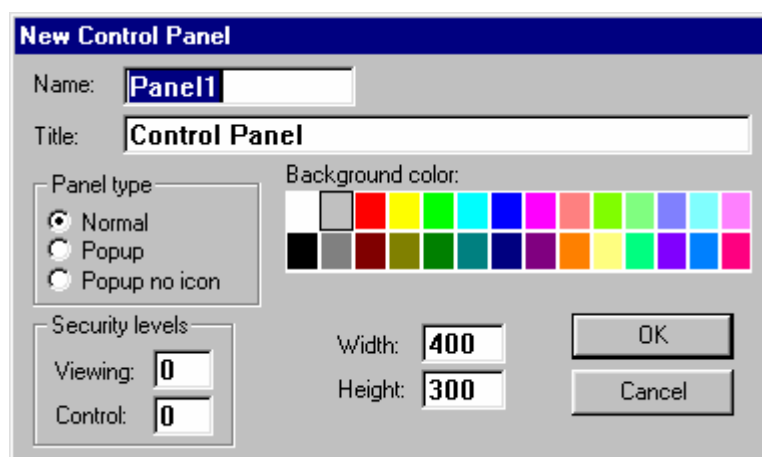
Citadel Database:

- ☒ Use Default Values

Computer Name:

Citadel Database Folder:

Un cop introduït el nom del nostre aplicatiu, clicar sobre OK. Ens apareixerà una nova finestra en la qual haurem d'indicar com s'anomenarà el nostre sistema que controlarem, i quina part anirem a crear. En el nostre cas creem el Panel1 que tindrà per títol Control Panel (que són els valors per defecte del programa).



New Control Panel

Name:

Title:

Panel type:

- ☒ Normal
- ☐ Popup
- ☐ Popup no icon

Background color:

Security levels:

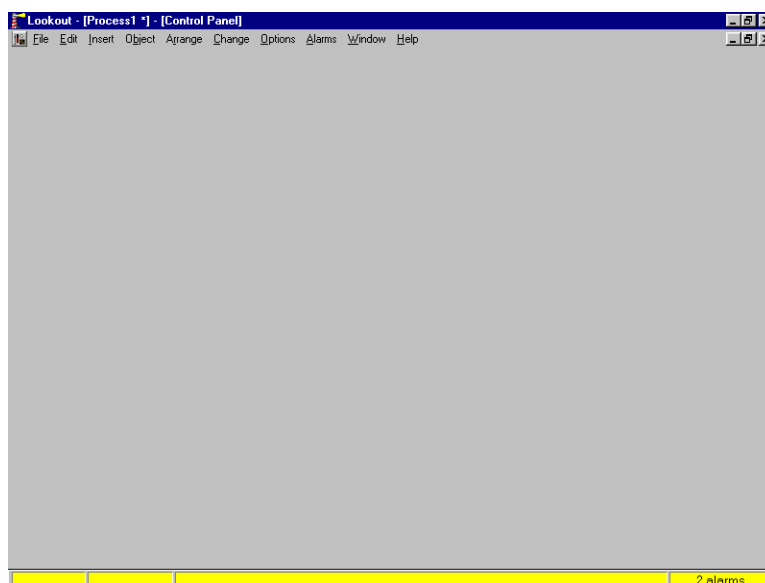
Viewing:

Control:

Width:

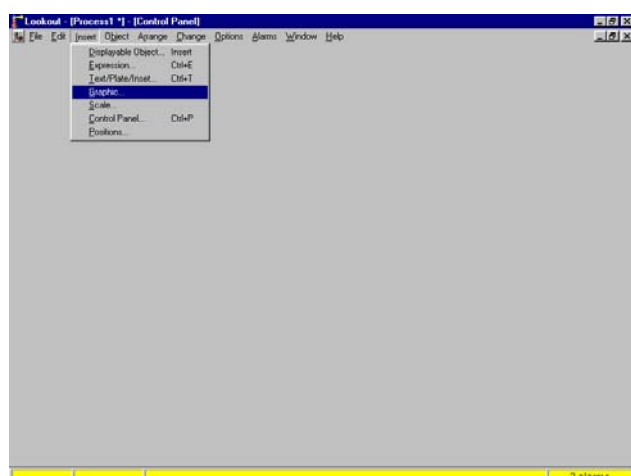
Height:

També podrem escollir les dimensions que tindrà aquesta aplicació com el color de fons que li volem donar. Clicar sobre OK i ens apareix la pantalla on nosaltres posarem els diferents objectes.



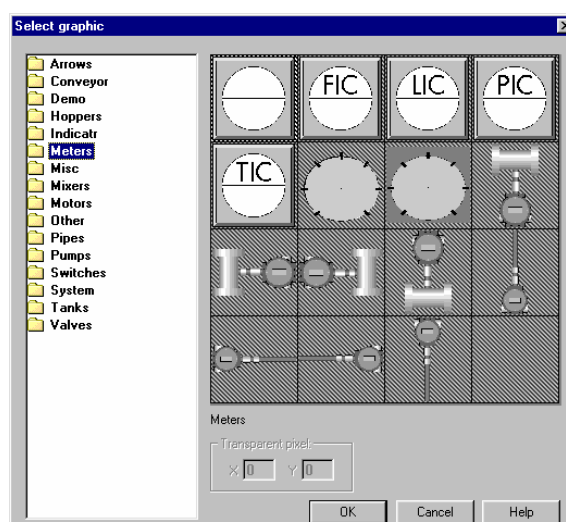
7.1.3. Introducció d'objectes fixos.

Clicquem sobre **Insert/Graphics...** i ens apareix una finestra amb molts arxius d'imatge que el programa porta per defecte. En el nostre cas, em hagut de crear una nova llibreria d'imatges per poder fer l'aplicació sobre la MPS1 (prèviament guardada en la carpeta Graphics que hi ha en el directori on està instal·lat el programa LookOut).



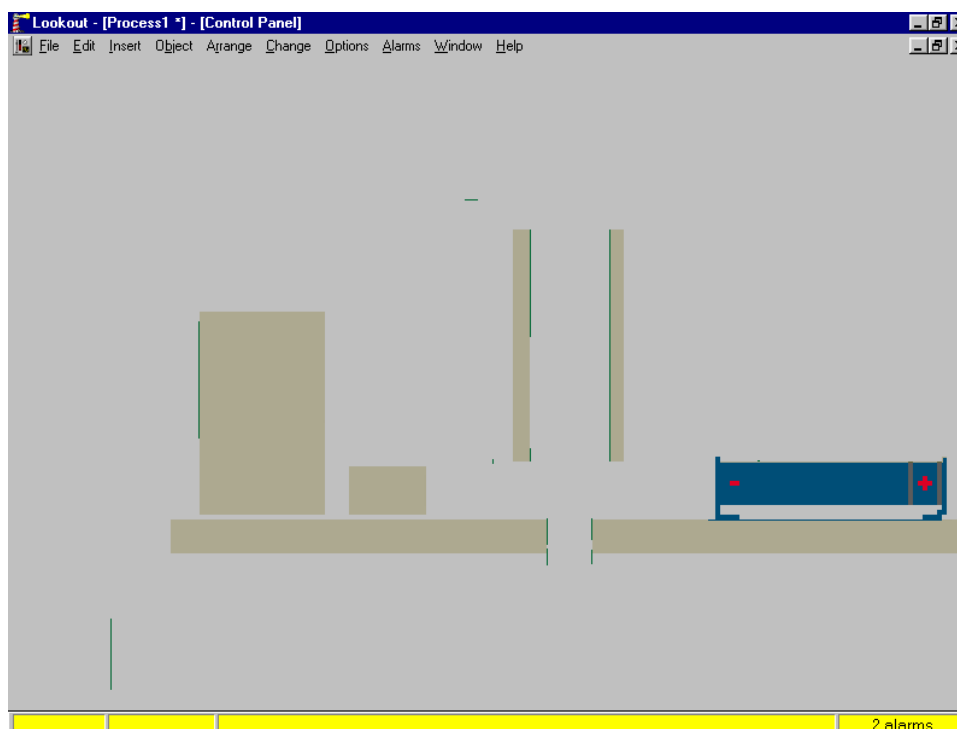
Totes les imatges que s'introdueixen a partir de la inserció de gràfics seran imatges en que no es podran realitzar animacions, es a dir, sols seràn imatges.

Observem navegant per les diferents carpetes, les imatges que nosaltres podem insertar.



En el nostre cas insertarem part de la MPS1 retocada sense les parts mòvils que s'hauran d'introduir com a animacions depenent dels senyals d'entrada (que s'explicarà més endavant).

Per introduir la MPS1 anem en aquesta mateixa pantalla, en la carpeta **Other** i escollim la imatge que ens interessa:



Per cada element que no volem que canviï mai, realitzarem el mateix procés.

Imatges que mai canvien són també les etiquetes de text. Per introduir etiquetes de text també fem servir un sistema semblant, però seguint diferents passos:

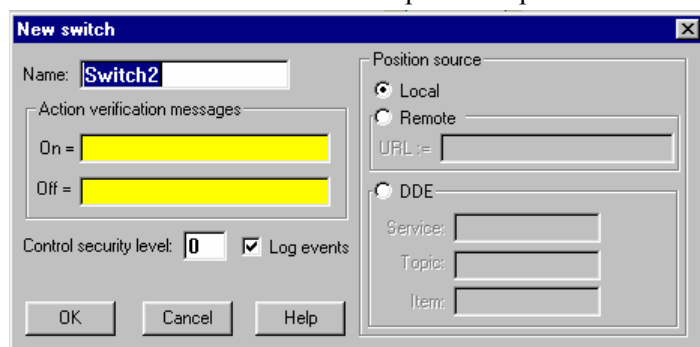
Menú **Insert** i **Text/Plate/Inset...** i ens apareixerà una pantalla on podrem escriure el text de la nostra etiqueta, com el color, font i fons que volem que tingui.

7.1.4. Inserció de parts mòvils o canviants.

7.1.4.1. Comandaments: interruptors i polsadors.

En primer lloc, abans d'introduir qualsevol part mòvil, em d'introduir el comandament o comandaments que controlaran les diferents accions sobre les parts que nosaltres volem animar.

Per crear doncs un interruptor o un polsador em d'anar a menú **Object/Create...** i sortirà una finestra



on trobem varies carpetes que contenen elements que podem crear, com per exemple polsadors (**pushbutton**), interruptors (**switch**), potenciómetres (**pot**), ... de la carpeta **display**.

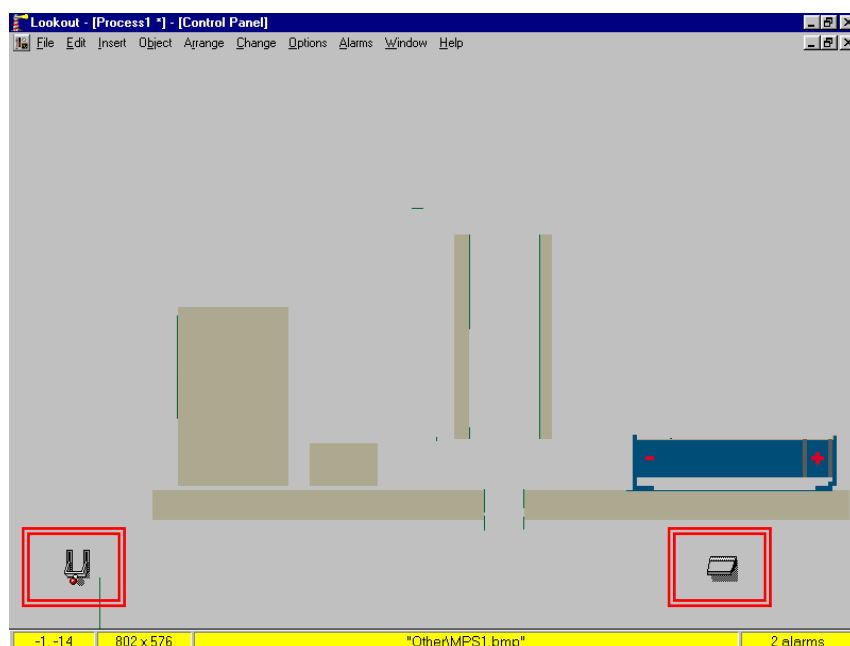
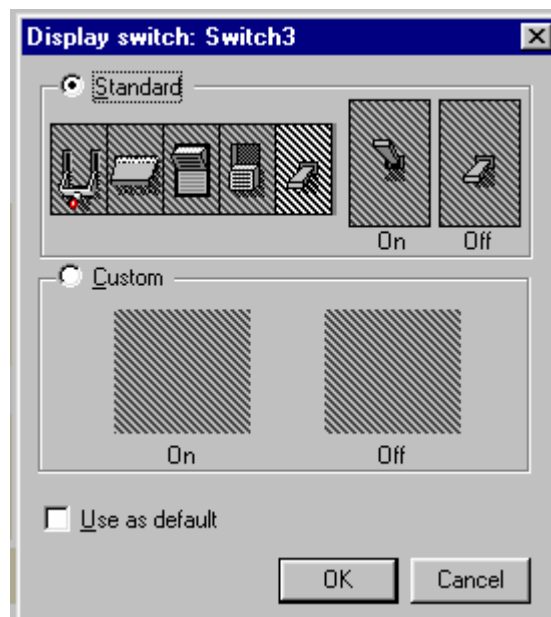
Podem començar per introduir l'interruptor general de la nostra simulació. Seguint l'explicació anterior doncs, anem a realitzar l'inserció d'aquest element, que serà un switch.

Sortirà una pantalla en la que és molt important el nom que tindrà el nostre interruptor, perquè el senyal que introdueix aquest és el propi nom, i haurèm de treballar amb ell. També trobem una regió amb el

fons de color groc, que són els missatges de verificació, és a dir, que si volem relitzar alguna acció sobre aquest element, preguntarà amb un d'aquests missatges si volem realitzar l'acció o no. Si ho deixem en blanc, no sortirà missatge de verificació i realitzarà l'acció directament. Cliquem OK i surt una nova finestra que serà l'estil que volem que tingui el nostre interruptor. Hi han varis tipus de interruptors que el programa té per defecte, però podem crear nosaltres un nou tipus d'interruptor, clicant sobre la corresponent zona. Si sel·leccionem **custom**, clicant sobre la pantalla ratllada, ens apareix una pantalla on podem escollir quin tipus de gràfic volem que tingui l'interruptor en ON i quin tipus de gràfic volem que tingui l'interruptor quan està en OFF.

Cliquem sobre OK, i ja tenim l'interruptor creat. Només cal moure aquest en la posició que volem que estigui en la pantalla.

Per crear l'interruptor que controlarà el cilindre realitzem els mateixos passos, excepte l'estil que tindrà aquest interruptor.



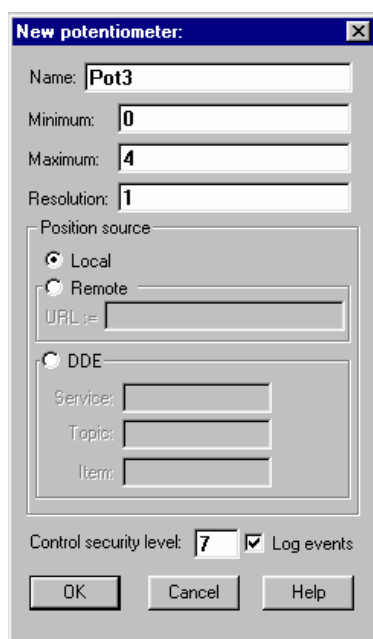
Un cop em creat el dos interruptors podem crear de la mateixa forma un pulsador (**pushbutton**) i també el col·loquem en la posició que volem.

Cal dir que quan creem un pulsador, l'estil amb que el creem és com una etiqueta, per lo que és bastant important els colors que utilitzem a l'hora de definir l'estil que volem que tingui el pulsador perquè es diferenciï de les possibles etiquetes que posem en pantalla.

7.1.4.2. Inserció de potencímetres.

Per poder controlar la posició del nostre manipulador, em creat un potencímetre. És interessant explicar com em creat aquest perquè pot ser una solució important per la creació d'objectes mòbils com a potencímetres que donen valor a alguna variable.

Per insertar-lo fem Menú **Object/Create...** i busquem en la carpeta **display** el sistema **Pot** i cliquem sobre ell i després OK. Apareix la següent pantalla:



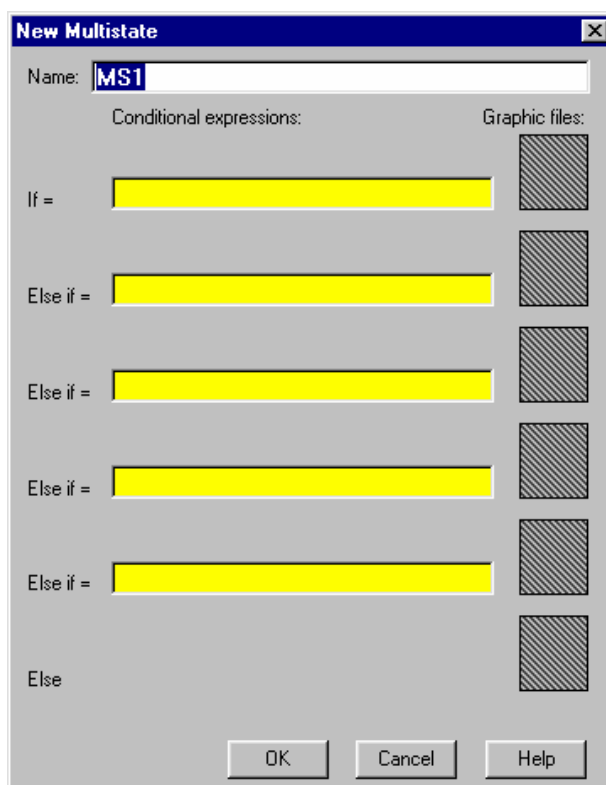
En el nostre cas, el manipulador tindrà 5 posicions:

1. Posició 0°.
2. Posició 45°.
3. Posició 90°.
4. Posició 135°.
5. Posició 180°.

Per lo que haurem de posar un mínim de 0 i un màxim de 4 que seràn les cinc posicions que em definit anteriorment, amb una resolució de 1 perquè volem que augmenti de 1 en 1 al moure aquest potencímetre, es a dir que cada vegada que movem el potencímetre passarà pels estats 0, 1, 2, 3 i 4. Cliquem OK i sel·leccionem l'estil que tindrà aquest potencímetre. I el coloquem en pantalla.

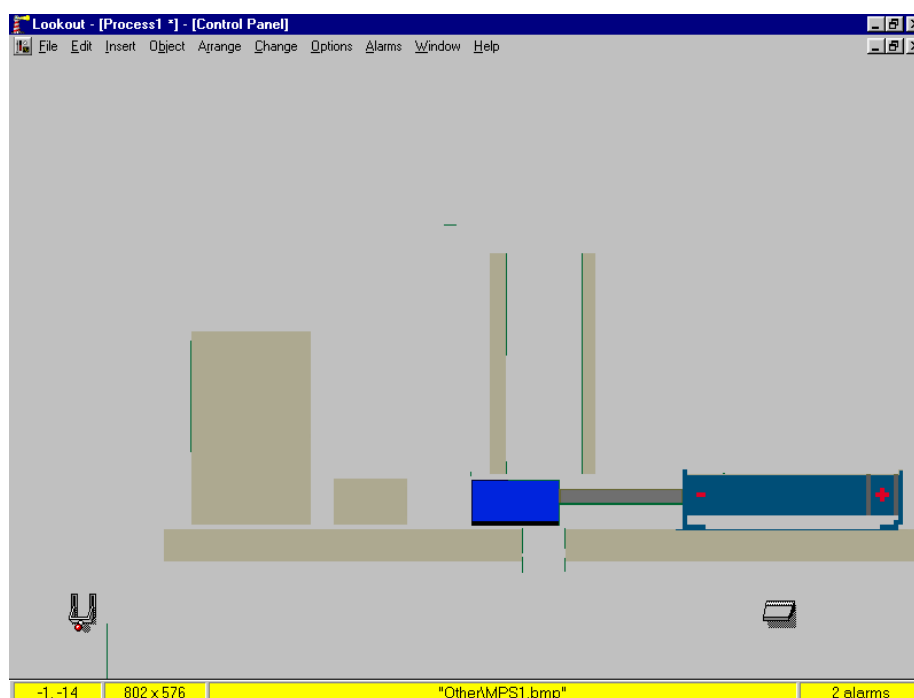
7.1.4.3. Inserció d'actuadors.

Per la inserció dels diferents actuadors fem servir la mateixa filosofia anterior, però en comptes d'insertar un interruptor, em d'insertar un **Multistate**. Ens apareix una nova finestra en la que em d'introduir com s'anomenarà aquest element i diferents imatges que volem que apareguin per pantalla segons les diferents condicions per les que volem que el nostre sistema canviï. Les condicions que no utilitzem, en aquestes les deixem en blanc i no apareixeran.

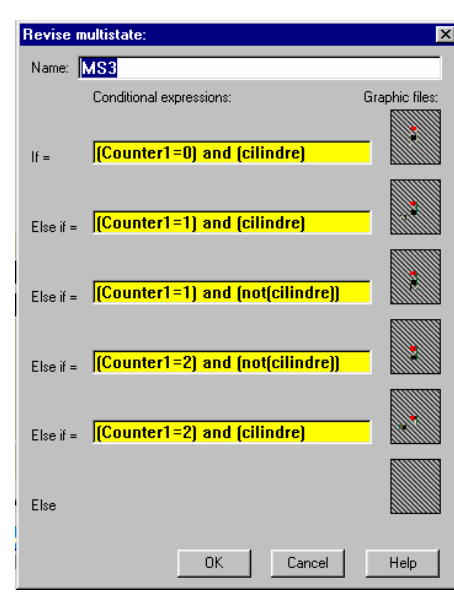
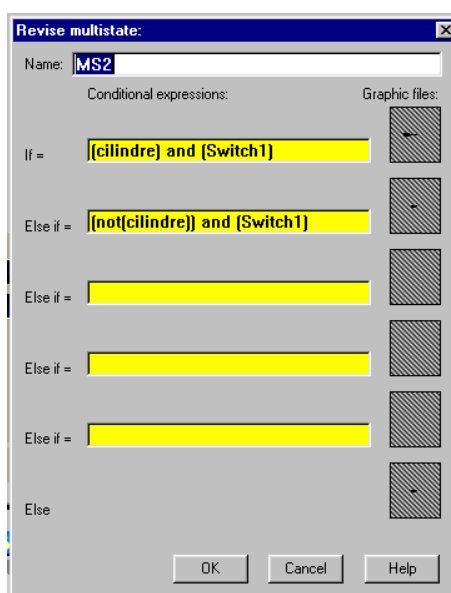
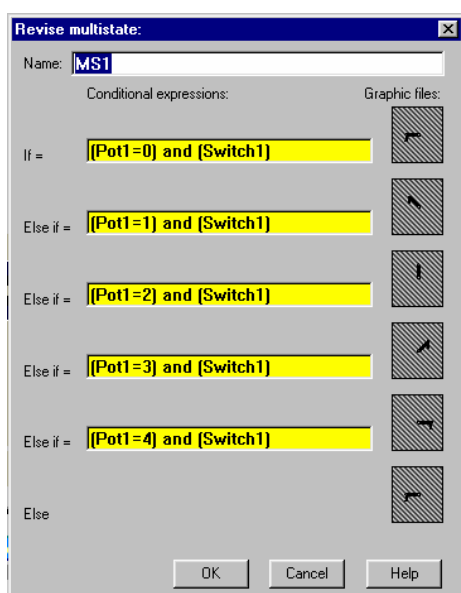


En el cas del cilindre, només hi han dues posicions a omplir que són quan el cilindre està extès o està en repòs. Per lo tant només tindrà dues condicions per als dos estats que té. Un cop posades les condicions cliquem OK i ja ens surt en pantalla la tija del cilindre. Ara el posicionem en el seu lloc i ja tenim el cilindre a punt.

Cal explicar que quan insertem un Multistate, les diferents imatges que anem posant sempre tenen el mateix emplaçament, per lo que si en algun dels estats ens apareix la imatge posicionada en un lloc erroni, em de modificar la imatge per al processador d'imatges com Paint, Coreldraw, ... i no per LookOut. Es a dir, representa que el Multistate que nosaltres em creat, per molt de que es composi per diferents imatges, es representa com si fos una sola imatge.



Amb la mateixa filosofia es crea el manipulador i els estats de les diferents peces. Veiem els diferents **multistates** que em creat:

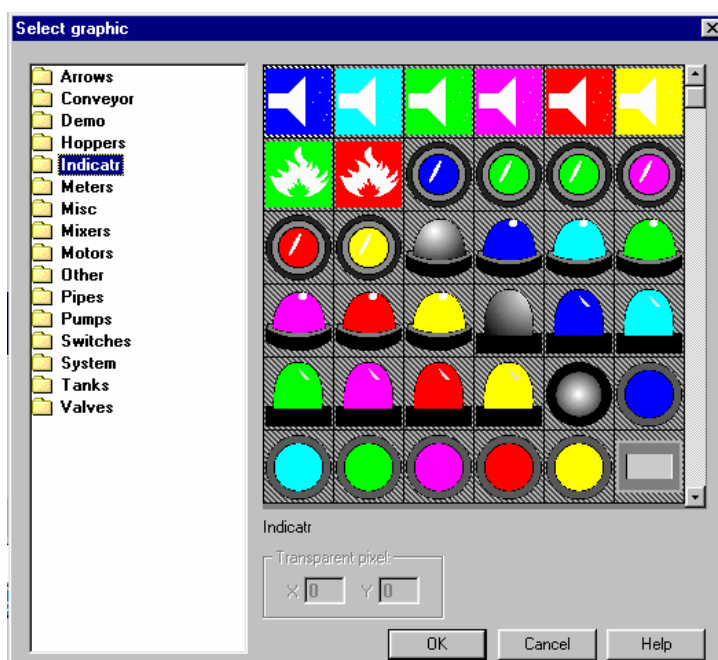


Creant tots aquests amb les condicions que hi hem posat controlem tota la planta MPS1. Ara només falta posar led's indicadors que ens indiquin l'estat en que es troben els diferents actuadors, que explicarem en el següent apartat.

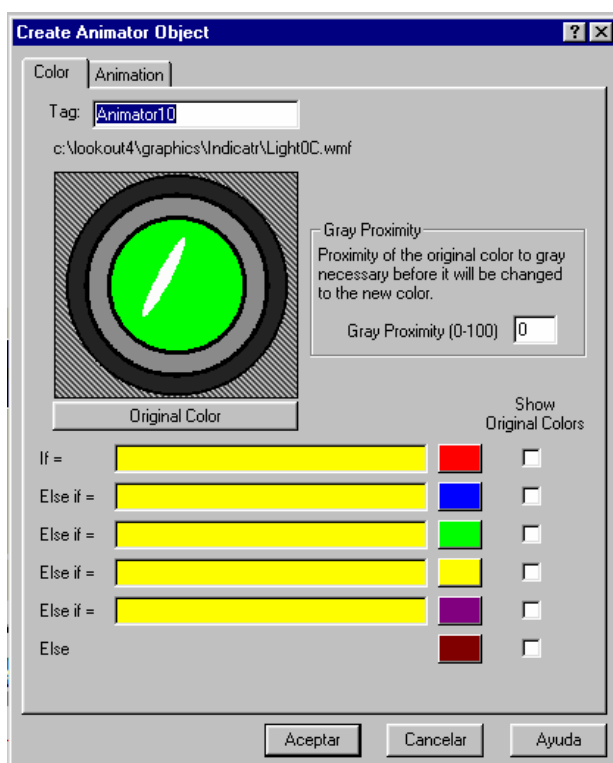
7.1.4.4. Indicadors.

Per crear led's indicadors per poder observar visualment millor l'estat dels diferents actuadors continuem fent servir la mateixa filosofia explicada en els apartats anteriors, però fent el següent: **object/create.../ display/animador**.

Un **animador** permet crear gràfics animats en els que canvia el color segons les diferents condicions que li posem. Per crear led's indicadors, els gràfics que seleccionarem seran els de la carpeta **indicatr**, que són els indicadors que el programa porta per defecte. I en seleccionem un d'ells:



Quan em seleccionat un d'ells, cliquem OK i sortirà la següent pantalla:



Tornem a trobar una finestra on posarem el nom que tindrà l'indicador i una zona amb diferents condicions que li podrem posar.

A la dreta de cada condició trobem uns requadres de colors. Aquests colors seran els diferents colors en que canviarà el nostre led segons les condicions que es vagin complint. Si volem canviar el color, només em de clicar damunt d'aquet requadre i ens sortirà una paleta on podrem canviar el color.

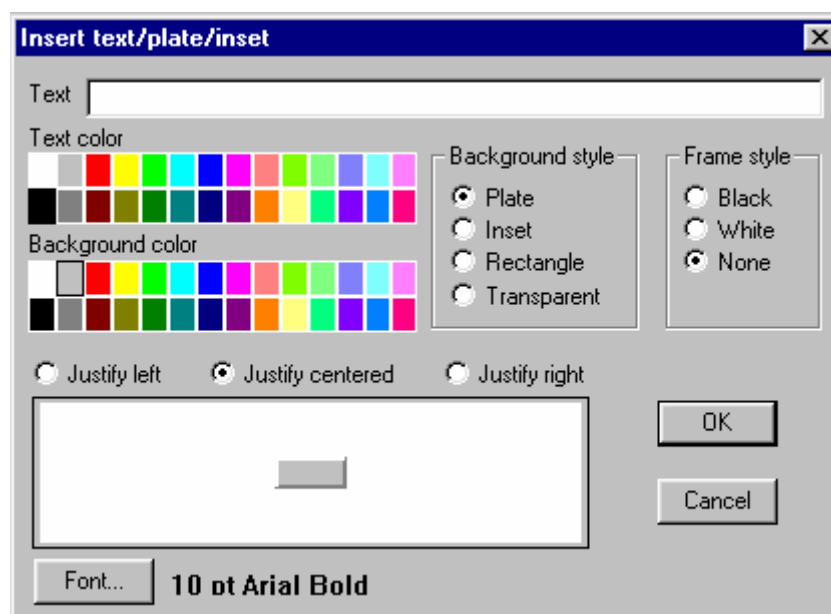
A més podem seleccionar una sèrie de quadres per si volem fer servir el color original del gràfic que em seleccionat, encara que hi hagi un altre color en el requadre. Per exemple, si observem la imatge, em seleccionat un indicador de color verd. En la primera condició hi ha seleccionat el color vermell. Si cliquem en la pestanya de **Show Original Colors** de la primera condició, en contres de visualitzar-se en roig, es visualitzarà amb el color original del gràfic que és el verd.

I així amb tots els indicadors que em introduït, però cada un amb les seves condicions. I col·loquem cada indicador allí on el volguem posar.

7.1.4.5. Inserció d'etiquetes.

Per acabar tota la simulació de la planta MPS1 i per saber els diferents interruptors a que fan referència, introduïm etiquetes de text que ens indicaran que és cada interruptor o indicador si cal.

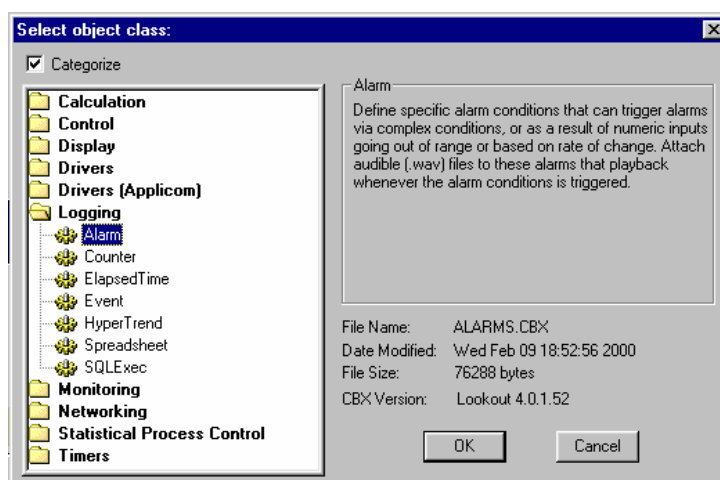
Per fer-ho em de pulsar sobre menú **Insert** i **text/plate/inset...**



Introduïm el text de la nostra etiqueta i seleccionem el color de la lletra i el color del fons de l'etiqueta, igual que l'estil que volem que tingui. Cliquem OK i posem l'etiqueta allí on la volem posar.

7.1.4.6. Inserció d'alarmes.

Per insertar una alarma em de clicar a Menú **Object/Create...** i buscar en la carpeta **Logging** el sistema Alarm. I cliquem OK.



Ens apareix la pantalla de generació d'alarmes:

En aquesta trobem quin nom volem posar a l'alarma que volem crear, el missatge que volem que aparegui en la pantalla d'alarmes i la condició que fa que s'activi aquesta alarma.

També apareix l'àrea de l'alarma, i aquí seleccionem la regió de LookOut.

També podem indicar quin nivell de prioritat té l'alarma, per si s'activen diferents alarmes alhora, quina volem que sigui la més important a visualitzar-se primer.

I si ens permet el PC la possibilitat d'emetre AUDIO, li podem assignar un senyal acústic a les alarmes que creem.

Les alarmes que es van activant es visualitzaran en la següent pantalla:

AlarmsWindow					
Time	Process	Area	Prior...	Object Name	Description
22:13:27.47	\\Jordi216\Process1	Lookout	4	\\Jordi216\Process1\...	No es pot carregar l'alimentador
22:12:10.47	\\Jordi216\Process1	Lookout	4	\\Jordi216\Process1\...	No es pot carregar l'alimentador

Podem visualitzar a quina hora s'ha generat l'alarma, en quin procés i en quina regió d'aquest procés s'ha produït. Quina prioritat té i quin és el missatge que ens identifica l'alarma.

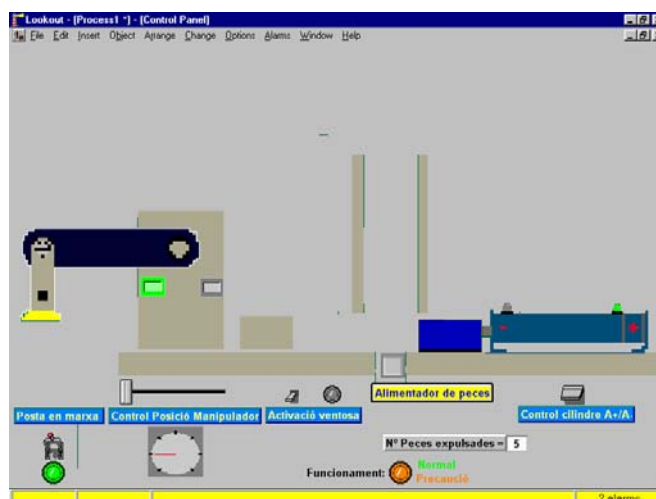
Si aquesta pantalla no surt, pulsar les tecles **Ctrl+A** i apareixerà aquesta finestra d'alarmes.

7.1.4.7. Posada en marxa de l'aplicació.

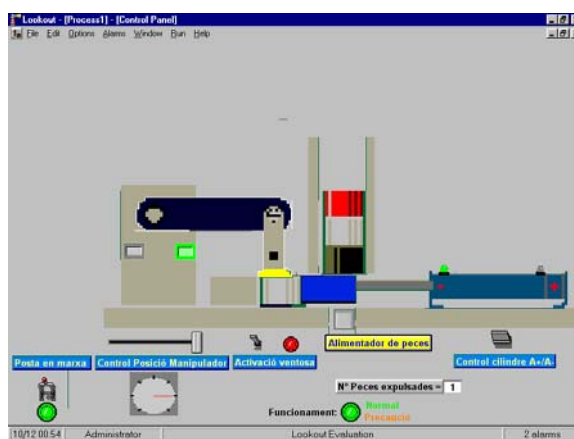
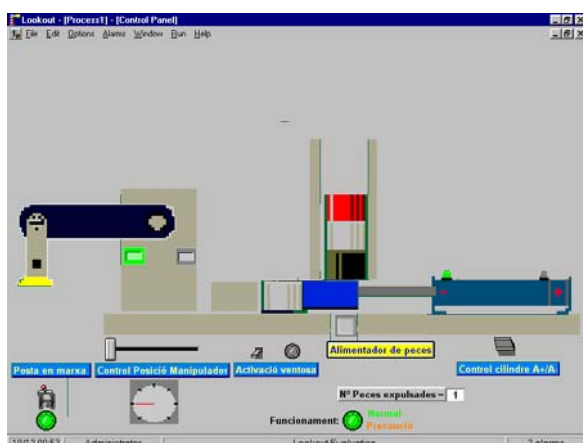
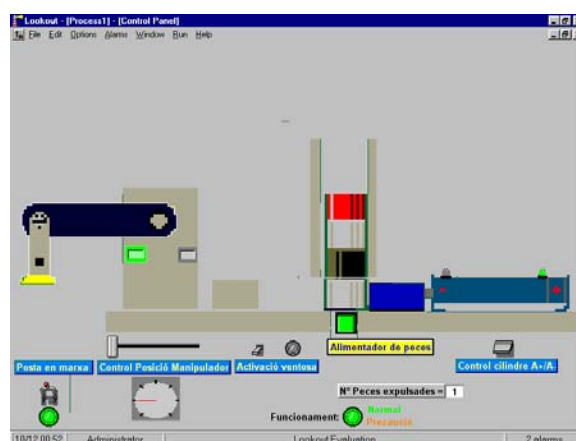
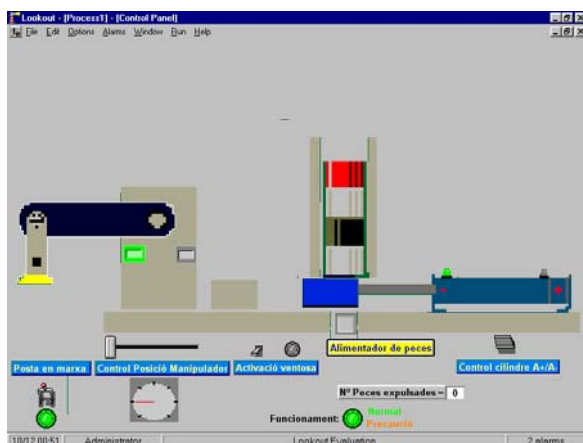
Un cop introduïts tots aquests elements explicats anteriorment, posem en marxa l'aplicació. Per passar a mode run el programa, pulsar **Ctrl+L** i en la pantalla que apareixerà clicar OK.

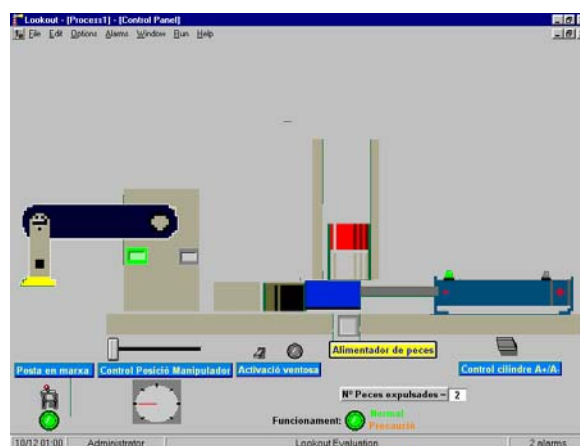
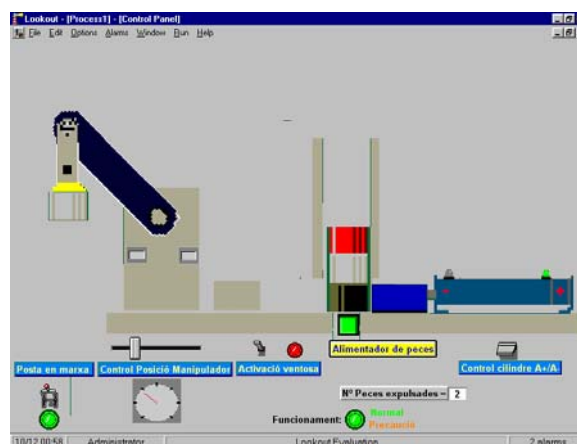
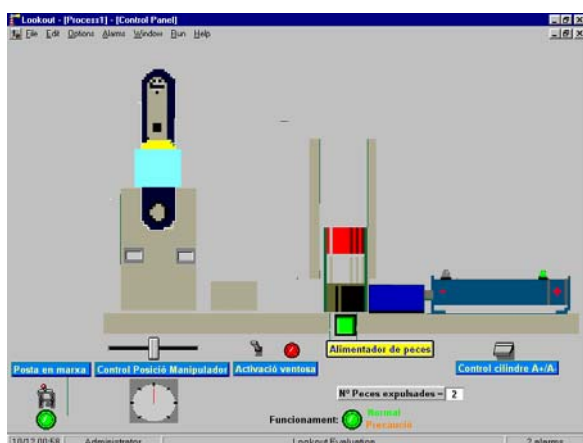
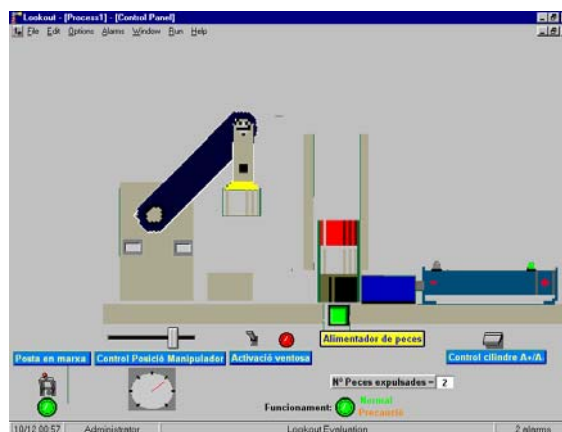
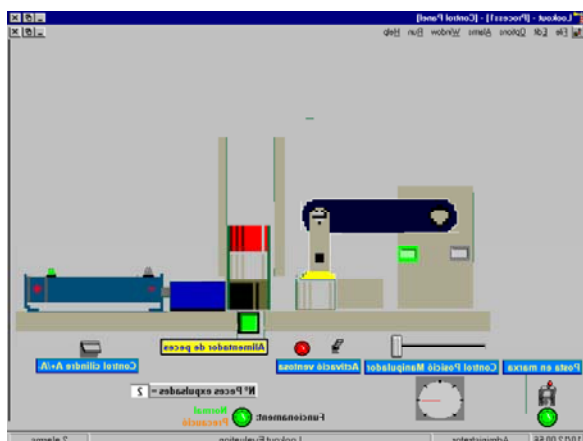
Si per qualsevol cosa volguessam modificar algun dels elements o insertar nous elements, pulsar **Ctrl+Space** i passarem al mode de treball.

La nostra aplicació final queda de la següent forma que es mostra en la següent imatge:



Només explicar que quan l'interruptor de **posta en marxa** està en OFF, no funciona cap element d'aquesta planta, i tots els indicadors estan apagats. Un cop posem el sistema en On, s'activa tot el sistema i podem passar a controlar-lo. Veiem les diferents seqüències creades:

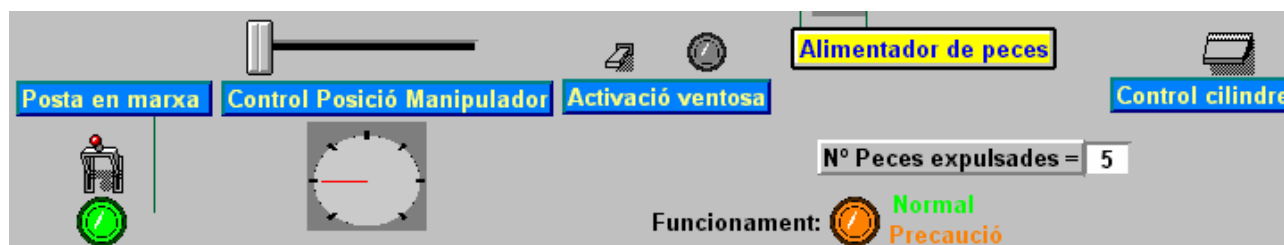




I així fins a expulsar totes les peces que hi ha a l'alimentador.

Quan s'acaben les peces de l'alimentador, observem que hi ha un led amb l'etiqueta funcionament, que canvia a taronja (precaució), que ens indica que el sistema funciona però està funcionant amb alguna anomalia. A la vegada hi ha una alarma que ens indica que no queden peces a l'alimentador.

Observem millor el panell de comandament de la nostra aplicació:



8. Conclusions.

En la realització d'aquest treball ens a permès conèixer l'aplicació de sistemes SCADA, es a dir, quins elements són necessaris per crear aquests tipus de sistemes, igual amb els conexionat i quines regles s'han de seguir per obtenir una bona realització d'aquests tipus de sistemes.

Pel que fa a la generació d'un aplicatiu amb un software SCADA, en el nostre cas LookOut de la casa National Instruments, em pogut adquirir el coneixement bàsic per la realització d'un sistema SCADA amb aquest software. Cal esmentar que per arribar a la realització d'un sistema automatitzat en el camp de la robòtica, ha estat una tasca difícil d'aconseguir ja que els manuals de LookOut i el propi software que em utilitzat està encarat en el control de processos continus en el temps, com poden ésser el control de temperatura, cabal de flúids, pressió, ... i s'ha hagut de crear una nova llibreria d'imatges i emmagatzemar-la en el directori on està instal·lat el programa LookOut aquesta nova llibreria que em hagut de crear nosaltres, que inclou imatges retocades del programa VEEP.

També em arribat a la conclusió, que a l'hora d'escollir el PLC a utilitzar en el control de la nostra planta o plantes en general i després realitzar el sistema SCADA d'aquest, és molt important relitzar un estudi econòmic i funcional del PLC a implementar, ja que si per exemple utilitzem un PLC de la casa OMRON, aquest fa servir el protocol de comunicacions HOST-LINK, per lo que podem connectar directament el PLC amb el *port sèrie* del nostre PC, i si utilitzem un PLC de la casa SIEMENS, el protocols que utilitza aquest són protocols PPI i MPI però utilitzant unes tarjes d'adquisició de dades Applicom, i aquestes econòmicament tenen un alt cost. Per lo que si valorem entre un sistema o l'altre, amb el PLC d'Omron, utilitzant el driver d'aquesta casa ja podem realitzar directament un sistema SCADA que controli aquest tipus de sistema, ja que no necessita de cap interfície de connexió entre PC i PLC.

9. Bibliografía.

- LookOut Evaluation Guide.
- LookOut Process Control Software System.
- www.ni.com/lookout
- www.ni.com/labview
- www.geocities.com/ResearchTriangle/3139/scada.html
- <http://isa.uniovi.es/genia>
- www.energytec.com.mx/sisscada.html
- www.monografias.com/trabajos11/sisco/sisco.shtml
- www.efalcom.com/scada.html
- www.instrucontrol.com/Foundation_Fieldbus_Seminar/sld001.htm
- www.automatas.org/redes/scadas.htm
- www.tecsup.edu.pe/capacitacion/index.php
- Fascicles revista “Automática e instrumentación”.